Universitat Politècnica de Catalunya
Programa de Doctorat de Matemàtica Aplicada

Departament de Matemàtica Aplicada III

NURBS-Enhanced Finite Element Method (NEFEM)

by Rubén Sevilla

Doctoral Thesis
Advisors: Antonio Huerta
Sonia Fernández-Méndez

Barcelona, July 2009

Contents

List of Figures

Chapter 1

Introduction

Several authors have pointed out the relevance of an accurate representation of curved boundaries in many computational mechanics problems in the last years, and have proposed different alternatives to implement an accurate or exact boundary representation. This thesis proposes a novel approach, the NURBS-enhanced finite element method (NEFEM). It allows considering the exact geometric description of the domain, by means of the usual CAD boundary representation with NURBS. Interior elements (i.e. elements not having an edge or face in contact with the NURBS boundary) are treated as standard finite elements (FEs). Therefore, in the vast majority of the domain, interpolation and numerical integration are standard, preserving the computational efficiency of the classical finite element method (FEM).

This chapter describes the necessity of an accurate geometric description in the numerical solution of electromagnetic scattering and compressible flow applications. Several methodologies has been proposed in the literature to treat curved boundaries. This chapter presents a brief state of the art with a historical perspective. Finally, the objectives of the work are described and an overview of the thesis is presented.

1.1 The importance of the geometrical model in finite element simulations

The origin of the FEM is mainly attributed to pioneer works in the field of airplane structural analysis, see Clough (2004) for an overview of the early history of the FEM. The need of curved elements to improve the quality of the shape discretization soon arose. *Isoparametric elements* were the first approach introduced to efficiently deal with curved boundaries, see Zienkiewicz (1971). The key idea was to employ the same polynomial functions to approximate the solution and the geometry, hence the term *isoparametric*. This approach was rapidly adopted for solid mechanics applications due to its straightforward implementation and its relatively good performance.

Despite of this early introduction of curved FE techniques for solid mechanics applications, in the 1980s and 1990s the geometric description was an important handicap for computational fluid dynamics (CFD). In particular, linear approximation of curved walls in numerical solutions of Euler equations of gas dynamics was identified as the origin of spurious entropy production near curved boundaries. For discontinuous Galerkin (DG) methods, see Cockburn (2004), the problem was identified by Bassi and Rebay (1997). In a finite volume (FV) context (Barth and Ohlberger, 2004) the problem was identified by Barth (1998). This fact has motivated several enhancements of wall boundary condition in domains with curved boundaries, both in DG and FV methods, see the recent works by Krivodonova and Berger (2006) and Wang and Liu (2006) among others.

An accurate representation of the geometry is not exclusive of fluid mechanics. Maxwell's equations are also very sensitive to an accurate geometric description. Xue and Demkowicz (2005) study the error induced by isoparametric approximations of curvilinear geometries. They show for the 3D Maxwell's equations in a sphere that exact mapping of the geometry reduces the error in one order of magnitude compared to isoparametric elements. Similar conclusions are derived by Luo et al. (2001) for linear elasticity problems. They conclude that sizable errors are present in the numerical solution when the order of the geometric approximation is lower than the order of

the functional interpolation. The relevance of an accurate geometric model for some applications in solid mechanics is also illustrated by Muñoz (2008), where the use of B-splines is proposed for the geometric representation of the interface in frictionless contact problems.

These handicaps have motivated a considerable development of new curved FE techniques and several methodologies were proposed since the introduction of the isoparametric FEM, by the late 1960s. The following is a brief overview with a historical perspective.

The technique presented by Zlamal (1973a) is recognized to be the first FEM considering an exact boundary representation. Triangular elements with one curved edge were introduced, and the isoparametric mapping was modified to map a reference element into the triangular element with an exact boundary description. A similar approach was developed by Scott (1973), also using triangular elements with one curved side corresponding to the exact boundary. Alternatives to the standard polynomial approximation of the solution were also proposed within the context of curved FEs with an exact boundary representation, see for instance the rational basis by Wachspress (1973). Nevertheless, all these FE techniques with exact boundary representation were not a practical tool, but a mathematical idealization, due to the impossibility to extend the ideas to 3D domains. For instance, Zlamal (1973b) abandons the exact geometric concept to simplify the implementation; in fact, to enhance the approximation provided by isoparametric elements a polynomial approximation of the boundary is proposed.

Transfinite elements by Gordon and Hall (1973) represented an inflection point in the development of general procedures to exactly treat curved boundaries. The key idea was to introduce blending functions to define a mapping between a reference square and a subdomain with the boundary given by four parametric curves. Blending mappings were naturally adopted in the so-called p-version of the FEM (p-FEM), see Babuška et al. (1981) and a recent review by Szabó et al. (2004). The problem of geometric inaccuracies associated to the isoparametric transformation are removed by blending mappings, and high degrees of interpolation are successfully employed.

Efficient procedures to combine of h and p refinements were rapidly introduced in practical applications with particular emphasis on an efficient approximation of singular solutions, see Zienkiewicz et al. (1989) or Demkowicz et al. (1989).

Curved FE were developed during the 1970s without regarding the emerging *Computer Aided Design* (CAD) industry. The great impact of CAD technology in the FE community arrived in the 1980s. In fact, researchers on the field of shape optimization were the first to promote the so-called *marriage of CAD and FE*. In a shape optimization process, the integration of CAD into the analysis stage is crucial to avoid the geometric approximation inherent in a mesh. Schramm and Pilkey (1993) implement *transfinite elements* (Gordon and Hall, 1973) using non-uniform rational B-splines (NURBS) for the geometric description, see Piegl and Tiller (1995) . Obviously, the rational nature of NURBS leads to rational function maps between the reference element and the element in the physical space. Note however that the solution is approximated using polynomials. Some inaccuracies associated to the lack of satisfaction of the isoparametric concept are mentioned by Schramm and Pilkey (1993), but not further explained. To obtain an isoparametric approach, the exact boundary representation was again abandoned, and a simplification of the geometry with polynomial B-Splines was proposed. Thus, some of the advantages of NURBS, such as exact representation of conics, were lost.

Over the 1990s, other authors focused their attention on integrating NURBS technology into FE codes. For instance, Dey et al. (1997) introduce an element geometric mapping also based on blending functions with NURBS. More recently, Mäkipelto (2004), apply p-FEM with NURBS for the boundary representation to plane elasticity problems. The inaccuracies previously mentioned by Schramm and Pilkey (1993) are also reported. Moreover, the lack of satisfaction of the isoparametric concept is alleviated by a rational enrichment of the polynomial basis used to approximate the solution.

The relevance of an accurate geometry description also motivated, in the late 1990s, a new family of FE-like techniques based on CAD, which is still today object of intensive research: *isogeometric methods*. The key idea is to use the same CAD

representation for both geometrical design and analysis. Thus, contrary to classical FE methodologies, the whole domain is treated as a CAD entity, not only the boundary of the domain. Moreover, classical polynomial approximations of the solution are abandoned and the solution is approximated with the same basis used in the CAD environment. The first application is again encountered on shape optimization (Kagan et al., 1998), using B-Splines for the geometrical description and for the mechanical analysis. In the 2000s, more advanced CAD technology was applied following the same rationale. Cirak et al. (2000) use *subdivision surfaces* for thin shell analysis. Relevant advantages are found in this context due to the sensitivity of shells to an inaccurate geometric representation. More recently, NURBS have been used to develop isogeometric methods, see for instance Inoue et al. (2005) or Hughes et al. (2005). Inoue et al. (2005) apply NURBS to shape optimization process is addressed. Hughes et al. (2005) present a more general framework known as *isogeometric analysis*. This work is not only focused on the accurate representation of the geometry, but also in the possibilities of NURBS as a basis for the approximation. See more recent advances by Bazilevs et al. (2008), realizing the isogeometric concept with a novel CAD technology: T-Splines.

The main drawback of isogeometric methods lies in the necessity of a solid CAD modeler for 3D domains. In practice, CAD manipulators work with the so-called boundary representation, that is, a parametric description of the boundary of the domain. Therefore, in 3D domains the geometry is given by a parametric surface rather than a parametric solid. In particular, for exterior problems, such as electromagnetic scattering or external aerodynamics applications, the use of NURBS to approximate the solution far away from the obstacle is not justified, and an unnecessary cost is introduced.

1.2 Objectives and overview

This thesis focuses on the development of a new FE technique for an efficient treatment of curved boundaries, the NURBS-enhanced finite element method (NEFEM).

The exact CAD description of the geometrical model is considered, but only for the boundary of the computational domain. At elements intersecting the NURBS boundary specific interpolation and numerical integration are proposed and, at elements not intersecting the boundary classical FE are used, preserving the efficiency of the FEM. The goal is to offer an efficient and easy-to-implement integration of the exact NURBS boundary representation into a FE code. The implementation and application of NEFEM to problems demanding an accurate boundary representation is also a primary goal of this thesis. Electromagnetic scattering and compressible flow applications are selected to show the performance of NEFEM.

For this purpose, the following partial goals are considered:

1. **Development and implementation of NEFEM.** A primary goal of NEFEM is to work with the exact geometric model given by the NURBS boundary representation of the domain. The use of the classical polynomial approximation of the solution is maintained, preserving the classical FE convergence properties and allowing a seamless coupling with standard FE on the domain interior (in elements not affected by the NURBS boundary representation). Efficient strategies to define the interpolation and the numerical integration in curved elements are needed. Chapter 2 is devoted to the presentation of NEFEM in 3D domains. Technical details are given in Appendices A and B. The presentation of NEFEM in 2D domains can be found in Sevilla et al. (2008a), and the details on the numerical integration in Sevilla and Fernández-Méndez (2009).

2. **NEFEM applied to electromagnetic scattering.** The solution of Maxwell's equations is highly sensitive to the geometrical description. Chapter 3 is devoted to present the application of NEFEM electromagnetic scattering problems. Several examples in 2D and 3D are considered, including classical test cases, benchmarks for the validation of computational electromagnetic (CEM) codes and more challenging applications. The ability of NEFEM to compute an accurate solution with coarse meshes and high-order approximations is shown. In particular, the performance of NEFEM elements containing corner and edge singularities is investigated.

The application of DG methods to electromagnetic scattering problems is reviewed in Appendix D. An important aspect in the numerical solution of such problems is the absorption of the outgoing waves in the artificial truncated boundary. Two perfectly matched layers (PMLs) for the solution of electromagnetic scattering problems with high-order DG methods are reviewed and compared in Appendix E.

3. **NEFEM applied to compressible flow.** A critical issue in the numerical solution of Euler equations is a proper imposition of wall boundary conditions at curved boundaries. NEFEM is presented in Appendix F as a powerful method for the solution of compressible flow problems. NEFEM overcomes the problem of spurious entropy production at curved walls when linear elements are used. In addition, NEFEM is shown to be more competitive than the classical isoparametric FEM also for high-order computations. Further details and examples are presented by Sevilla et al. (2008b).

4. **Comparison.** There are several FE techniques for the treatment of curved boundaries. Nevertheless, there are important differences between these methodologies and NEFEM, both from a theoretical and from a practical point of view. A critical comparison between NEFEM and other curved FEs is presented in Appendix C. Moreover, through the thesis, several numerical examples show a performance comparison between NEFEM and other curved FEs, namely isoparametric FEM, cartesian FEM and p-FEM, reviewed in Appendix C. Comparisons are performed for second-order problems using a continuous Galerkin formulation, for electromagnetic scattering applications solved with a DG formulation, and for compressible flow problems also solved with a DG formulation.

Chapter 2

NURBS-Enhanced Finite Element Method (NEFEM)

The importance of the geometrical model in FE simulations has been pointed out by several authors, see among others Bassi and Rebay (1997); Cirak et al. (2000); Luo et al. (2002); Xue and Demkowicz (2005); Krivodonova and Berger (2006). Non-uniform rational B-splines (NURBS), see Piegl and Tiller (1995), are widely used for geometry description in CAD. This fact has motivated new numerical methodologies considering an exact representation of the computational domain with NURBS, such as the isogeometric analysis (Hughes et al., 2005) and NEFEM (Sevilla et al., 2008a).

In the *isogeometric analysis* the whole domain is represented as a NURBS entity, and the solution of the boundary value problem is approximated with the same NURBS basis used for the description of the geometry. There are two main differences between the isogeometric analysis and NEFEM. First, NEFEM considers the exact NURBS description for the boundary of the computational domain, the usual information provided by a CAD software. Secondly, NEFEM approximates the solution with a standard piecewise polynomial interpolation. Moreover, every *interior* element (i.e. elements not having an edge or face in contact with the NURBS boundary) can be defined and treated as a standard FE element. Therefore, in the vast majority of the domain, interpolation and numerical integration are standard, preserving the computational efficiency of the classical FEM. Specific numerical strategies for the interpolation and the numerical integration are needed only for those elements affected

by the NURBS boundary representation.

This chapter presents the fundamentals of NEFEM in 3D domains. Special attention is paid to the design of efficient strategies in order to define the interpolation and to perform the numerical integration on curved elements. Main ideas exposed in this chapter are complemented with technical details presented in Appendix B. Optimal a priori error estimates are given. Finally, second-order elliptic problems in 2D and 3D are used to test the performance of NEFEM in front of other curved FEs in a continuous Galerkin framework. The comparison in a DG framework is presented in Chapter 3 for solving electromagnetic scattering problems and in Appendix F for compressible flow applications. Appendix C presents a comparison from a theoretical and an implementation point of view.

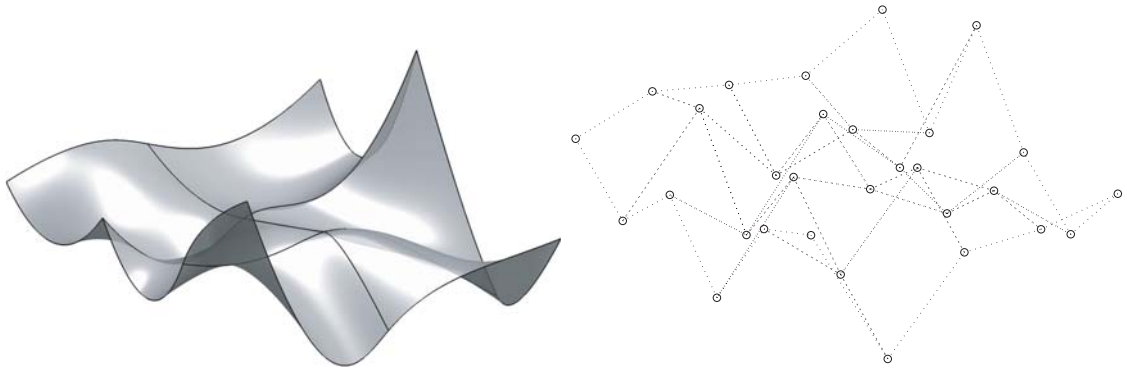
2.1 Preliminaries

Consider an open bounded domain $\Omega \subset \mathbb{R}^3$ whose boundary $\partial\Omega$, or a portion of it, is defined by NURBS surfaces. Every NURBS surface is assumed to be parametrized by

$$\boldsymbol{S} : [0,1]^2 \longrightarrow \boldsymbol{S}([0,1]^2) \subseteq \partial\Omega \subset \mathbb{R}^3.$$

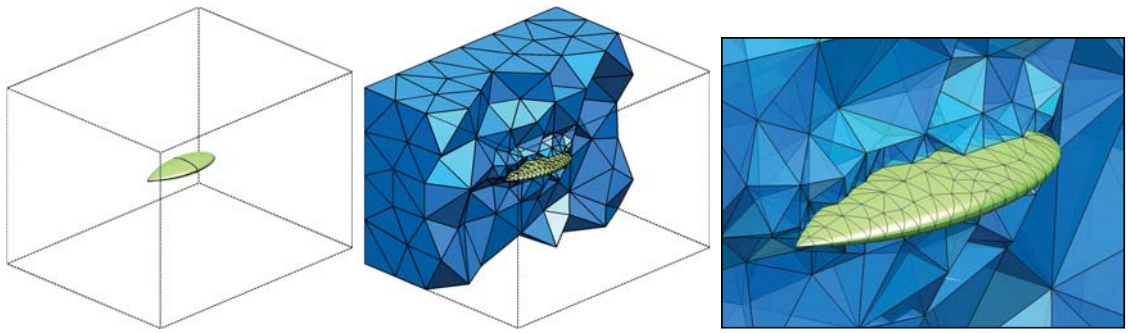
NURBS surfaces are piecewise rational functions defined in parametric form. They are expressed in terms of a *control net*, in which each point acts as an attractor of the resulting surface. See an example in Figure 2.1, where the lines over the surface represent changes of NURBS definition (*knot lines*). NURBS allow to represent a wide range of surfaces, and contrary to polynomial B-Splines or subdivision surfaces, they allow to represent conics exactly. Basic concepts on NURBS are recalled in Appendix A, see Piegl and Tiller (1995) for a complete presentation.

A regular partition of the domain $\overline{\Omega} = \bigcup_e \overline{\Omega}_e$ in elements is assumed, such that $\Omega_i \bigcap \Omega_j = \emptyset$, for $i \neq j$. For instance, Figure 2.2 shows a computational domain with part of the boundary defined by NURBS surfaces corresponding to the NASA almond, a useful geometry for benchmarking electromagnetic scattering codes, see



(a) (b)

Figure 2.1: (a) NURBS surface with knot lines and (b) control net



(a) (b) (c)

Figure 2.2: (a) Domain with part of the boundary defined by curved NURBS surfaces corresponding to the NASA almond, (b) cut through an unstructured tetrahedral mesh in blue, with the surface triangular mesh of the almond in green, and (c) detail of the mesh near the almond

Dominek and Shamanski (1990). A cut through an unstructured tetrahedral mesh is also represented in Figure 2.2, including the surface triangular mesh on the almond.

As usual in FE mesh generation codes, it is assumed that every curved boundary face belongs to a unique NURBS. That is, one element face can not be defined by portions of two, or more, different NURBS. Note however that the piecewise definition of NURBS is independent of the mesh discretization. Thus, the NURBS parametrization can change its definition inside one face, that is, FE edges do not need to belong to knot lines. This is a major advantage with respect to the isogeometric analysis

(a) (b) (c)

Figure 2.3: (a) Knot lines of the NURBS surfaces defining the NASA almond, (b) surface triangulation, and (c) surface triangulation and knot lines

of Hughes et al. (2005). Moreover, different faces of an element can be defined by different NURBS surfaces. Figure 2.3 shows the knot lines of the surfaces that define the NASA almond and the surface triangulation corresponding to the mesh of Figure 2.2. It can be observed that the spatial discretization is independent of the piecewise NURBS surface parametrization.

In NEFEM, the geometric definition of a curved element is given by the NURBS information of the boundary. For instance, Figure 2.3 (b) shows the tetrahedral faces on the NURBS boundary, and Figure 2.2 (c) shows some internal faces with an edge on the NURBS boundary. The formal definition of curved faces in a NEFEM tetrahedral mesh is given in this section.

Let Υ_e be a face on the NURBS boundary parametrized by $\boldsymbol{S}$, and $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3 \in \partial\Omega$ the three vertices on the NURBS boundary, see Figure 2.4. A straight-sided triangle Λ_e in the parametric space of the NURBS is uniquely defined by the parametric coordinates of the vertices, that is $\boldsymbol{S}^{-1}(\boldsymbol{x}_1)$, $\boldsymbol{S}^{-1}(\boldsymbol{x}_2)$ and $\boldsymbol{S}^{-1}(\boldsymbol{x}_3)$. The exact curved face in cartesian coordinates, Υ_e, is defined as the image of the straight-sided triangle Λ_e by the NURBS parametrization $\boldsymbol{S}$,

$$\Upsilon_e := \boldsymbol{S}(\Lambda_e), \tag{2.1}$$

as illustrated in Figure 2.4.

Interior curved faces with an edge on the NURBS boundary are defined as a convex linear combination of the curved edge and the interior face node. For instance, the

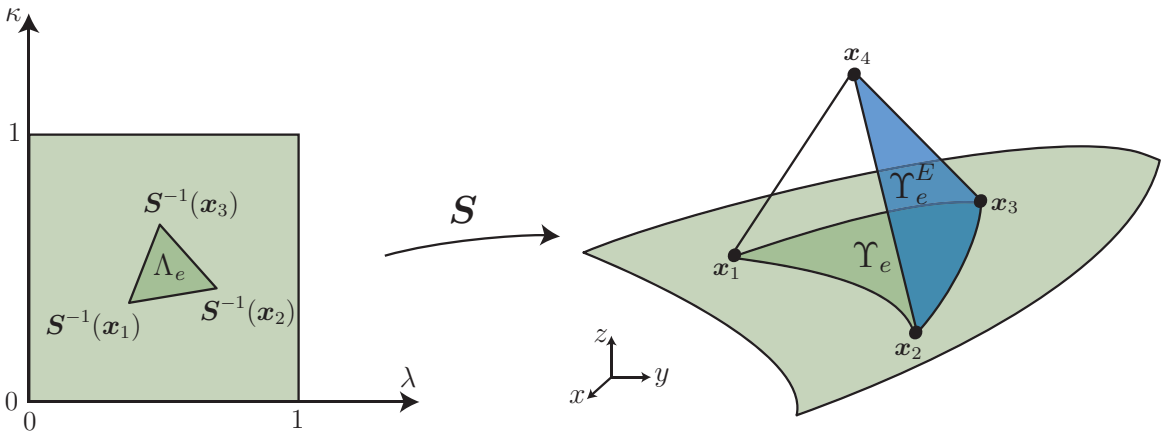


Figure 2.4: Definition of a curved tetrahedral face on the NURBS boundary, Υ_e, and a curved tetrahedral face with an edge on the NURBS boundary, Υ_e^E

curved face Υ_e^E represented in Figure 2.4 is parametrized by

$$\begin{aligned} \boldsymbol{\Theta} \ &: [0,1]^2 \longrightarrow \Upsilon_e^E \\ &(\varrho,\sigma) \longmapsto \boldsymbol{\Theta}(\varrho,\sigma) := (1-\sigma)\boldsymbol{\theta}(\varrho) + \sigma \boldsymbol{x}_4, \end{aligned}$$

where $\boldsymbol{\theta}$ is the parametrization of the curved edge containing vertices $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$.

Note that this approach to define curved faces ensures the same definition for an internal curved face viewed from the two elements sharing this face. Note also that other faces are present in real meshes, for instance faces with several edges defined by different NURBS. The definition of all possible curved faces is performed in Appendix B.

2.2 Polynomial interpolation

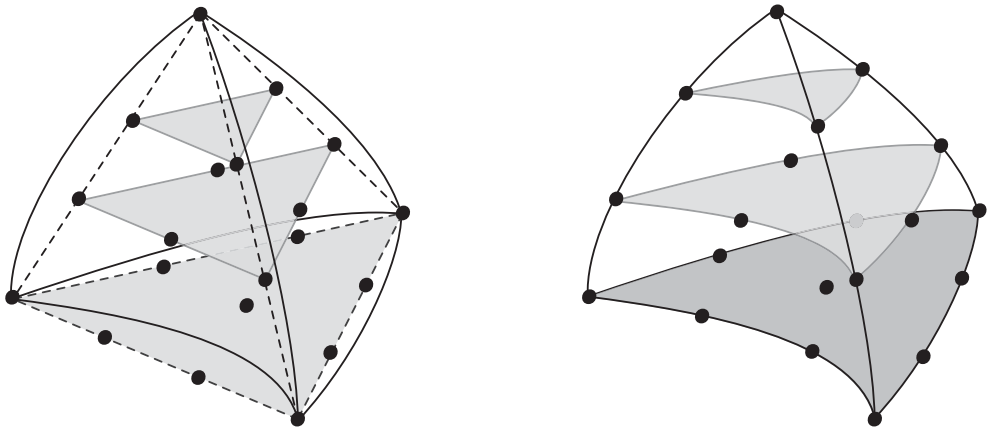
This section is devoted to highlight the main properties of polynomial interpolation in elements affected by the NURBS boundary representation. A complete analysis and discussion of the polynomial basis adopted in NEFEM is presented in Appendix B.

In order to work with standard polynomial approximations, Lagrange polynomials (that is, standard nodal interpolation) are considered. To ensure reproducibility of polynomials in the physical space, NEFEM defines the approximation directly with cartesian coordinates, $\boldsymbol{x} = (x, y, z)^T$, that is

$$u(\boldsymbol{x}) \simeq u^h(\boldsymbol{x}) = \sum_{i=1}^{\mathtt{n_{en}}} u_i N_i(\boldsymbol{x}),$$

where u_i are nodal values, N_i are polynomial shape functions of order p in $\boldsymbol{x}$, and $\mathtt{n_{en}}$ is the number of element nodes. Recall that this is not the usual approach to define a functional approximation in a FE context. For instance, in the classical isoparametric FEM, or in p-FEM, the approximation is defined with local coordinates, in a reference element. Then, local and cartesian coordinates are related using an isoparametric mapping or using blending mappings, see the discussion in Appendix C.

Different options can be considered for the definition of a nodal distribution in Ω_e. Any nodal distribution, such as equally-spaced nodal distributions, can be defined on the tetrahedral with planar faces given by the vertices of Ω_e, or adapted to the exact geometry, see Figure 2.5. The definition of a nodal distribution on the tetrahedral with planar faces, see Figure 2.5 (a), induces a marginal extra efficiency, because the nodal basis can be computed once for a reference element with planar faces and used for each curved element. Adapting the nodal distribution to the exact geometry, see Figure 2.5 (b), allows a seamless imposition of boundary conditions in strong form, directly imposing the value of the solution at nodes on the boundary. But, nodal distributions adapted to the boundary do not represent any implementation advantage if boundary conditions are imposed in weak form, as usual in DG formulations. Note however the evolution of the condition number, shown in Figure 2.6, for the element mass matrix as a function of the polynomial degree of the approximation. Adapted distributions of nodes induce an important reduction on the condition number. For high-order approximations equally-spaced nodal distributions lead to ill-conditioned matrices. In this case, specific nodal distributions should be implemented in Ω_e, see for instance the near optimal distributions proposed by Chen and Babuška (1996), Hesthaven



(a) (b)

Figure 2.5: Equally-spaced nodal distribution for $p = 3$ (a) defined using the tetrahedral with planar faces represented by discontinuous lines, and (b) adapted to the exact geometry

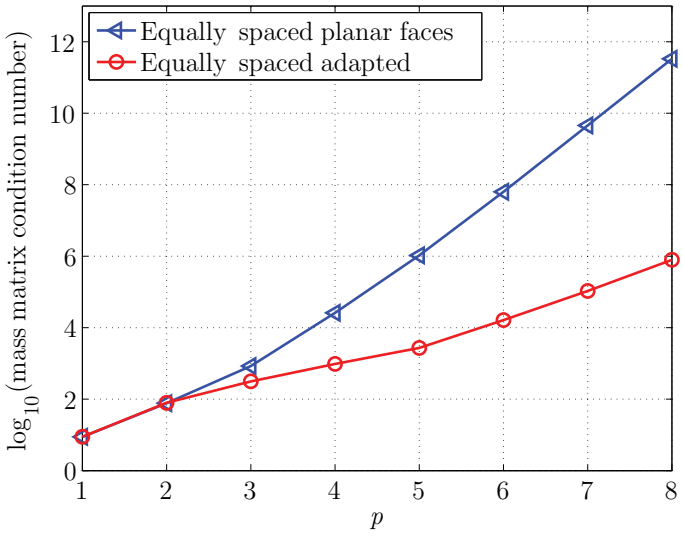
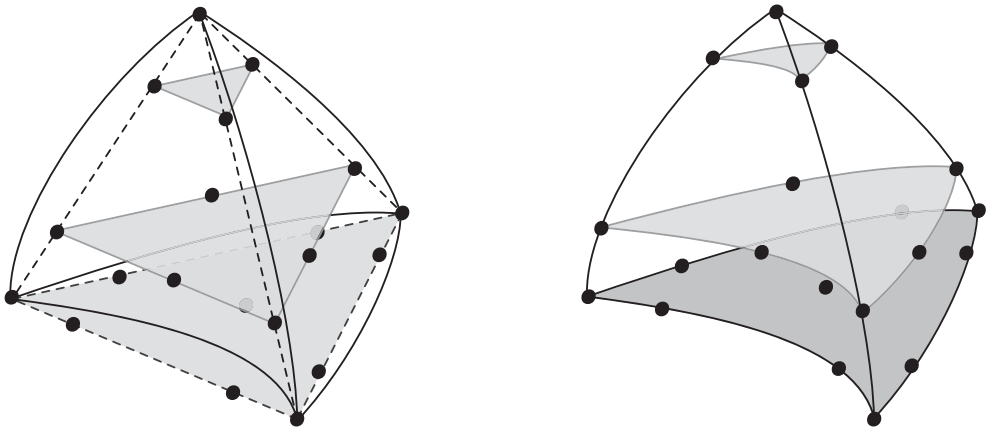


Figure 2.6: Condition number of the mass matrix as a function of the interpolation degree (p)

(2000), or Warburton (2006), for elements with planar faces. The adaptation of such distributions to the exact geometry may lead to an extra reduction in the condition number of the elemental matrices, see Sevilla et al. (2008a) for NEFEM in 2D. An example of a near optimal distribution in the tetrahedral element with planar faces



(a) (b)

Figure 2.7: Near optimal distribution proposed by Hesthaven (2000) for $p = 3$ (a) defined in the tetrahedral with planar faces represented by discontinuous lines, and (b) adapted to the exact element

and adapted to the exact geometry is represented in Figure 2.7, corresponding to the distribution proposed by Hesthaven (2000) for $p = 3$.

2.3 Numerical integration

The weak form of the problem requires both integrations over element faces and in the element interiors. All integrals in elements not having an edge or face in contact with the NURBS boundary are computed using standard procedures. For an element Ω_e affected by the NURBS boundary representation, it is necessary to design specific quadratures. Two cases must be considered, surface integrals (usually related to the implementation of natural boundary conditions or to flux evaluation over the face in a DG context) and volume integrals (standard integrals in the element Ω_e). As discussed in the previous section, since NEFEM uses polynomials to approximate the solution, the difficulties in numerical integration are only restricted to the definition of a proper numerical quadrature in a curved element, Ω_e, and a curved face. Special attention must be paid to the definition of suitable quadratures accounting for changes of NURBS definition within an element face or edge.

This section describes in detail the definition of numerical quadratures in a curved tetrahedral with a face on the NURBS boundary. For real applications it is necessary to design specific quadratures for other element typologies. For instance, an element with two faces defined by two different NURBS. Section B.2 presents an exhaustive description of numerical integration in 3D NEFEM.

Let Ω_e be an element with a NURBS face parametrized by $\boldsymbol{S}$, and $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3 \in \partial\Omega$ the three vertices on the NURBS boundary, see Figure 2.4. The curved face on the NURBS boundary is defined as the image of the straight-sided triangle Λ_e in the parametric space of the NURBS, see Equation (2.1). Therefore, a surface integral on a curved boundary face, Υ_e, can be written as

$$\int_{\Upsilon_e} f \, dx \, dy \, dz = \int_{\Lambda_e} f\big(\boldsymbol{S}(\lambda,\kappa)\big) \, \|J_{\boldsymbol{S}}(\lambda,\kappa)\| \, d\lambda \, d\kappa, \tag{2.2}$$

where f is a generic function (usually polynomial) and $\|J_{\boldsymbol{S}}(\lambda,\kappa)\|$ denotes the norm of the differential of the NURBS parametrization $\boldsymbol{S}$ (which, in general, is not a polynomial). An efficient option to evaluate integral (2.2) is to use a triangle quadrature (Wandzura and Xiao, 2003) in Λ_e. If changes of NURBS parametrization are present inside the parametric triangle Λ_e, the numerical quadrature must be designed to account for the piecewise NURBS parametrization. For instance, a triangulation of Λ_e such that each *subtriangle* has no changes of NURBS parametrization can be considered. Then, a different numerical quadrature is defined on each *subtriangle*, as represented in Figure 2.8.

For the computation of interior integrals, the element Ω_e is parametrized with the transformation

$$\begin{aligned} \boldsymbol{\Psi} \; &: \Lambda_e \times [0,1] \longrightarrow \Omega_e \\ &(\lambda,\kappa,\vartheta) \longmapsto \boldsymbol{\Psi}(\lambda,\kappa,\vartheta) := (1-\vartheta)\boldsymbol{S}(\lambda,\kappa) + \vartheta \boldsymbol{x}_4, \end{aligned}$$

where $\boldsymbol{x}_4$ denotes the internal vertex of Ω_e, see Figure 2.9. Then, the element integral

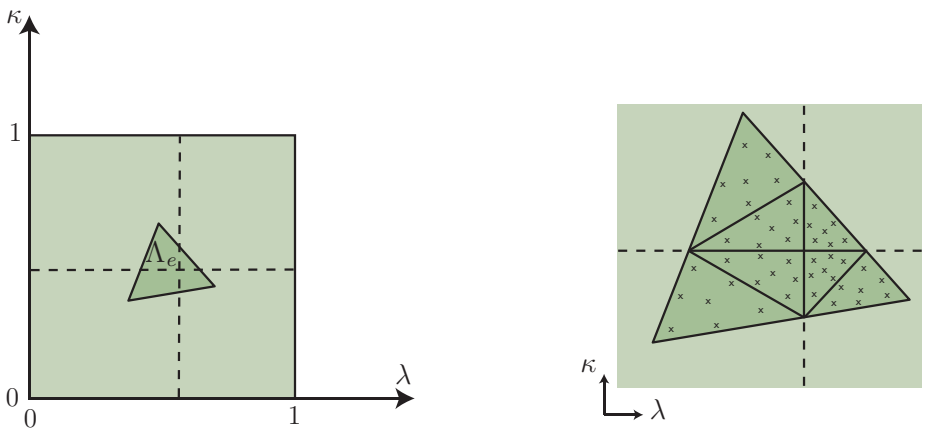


Figure 2.8: Definition of a numerical quadrature on Λ_e for the numerical integration on a curved tetrahedral face with changes of NURBS parametrization along discontinuous lines (knot lines): triangle in the parametric space and detailed view of the composite quadrature

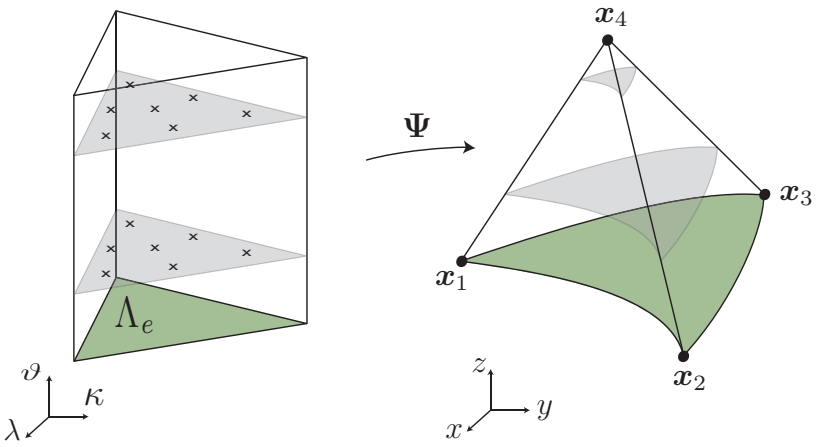


Figure 2.9: Transformation from $\Lambda_e \times [0,1]$ to Ω_e

is written as

$$\int_{\Omega_e} f \; dx\, dy\, dz = \int_{\Lambda_e} \int_0^1 f\big(\boldsymbol{\Psi}(\lambda,\kappa,\vartheta)\big) \; |J_{\boldsymbol{\Psi}}(\lambda,\kappa,\vartheta)| \; d\lambda \; d\kappa \; d\vartheta.$$

where f is a generic function (usually a polynomial), and $|J_{\boldsymbol{\Psi}}|$ denotes the determinant of Jacobian of the transformation $\boldsymbol{\Psi}$. A numerical quadrature on $\Lambda_e \times [0,1]$ is easily defined as a tensor product of a triangle quadrature in Λ_e and a 1D Gauss-Legendre

quadrature in $[0,1]$, see Figure 2.9. In fact, exact integration is feasible in the third parameter due to the linearity of $\mathbf{\Psi}$ with respect to ϑ. For a NEFEM solution with a degree of approximation p, exact integration is provided by a Gauss-Legendre quadrature with $p+2$ integration points. To account for changes of NURBS definition, only the quadrature in Λ_e must be modified, see Figure 2.8. This represents an advantage in front of the numerical integration strategy for the p-FEM with a NURBS boundary description, see more details in Section C.4.

If a DG formulation is adopted, it is also necessary to define numerical quadratures on internal faces of Ω_e, that is faces with one edge on the NURBS boundary. Such quadratures are described in Appendix B.

2.4 *A priori* error estimates

Since NEFEM considers standard FE polynomial interpolation, see Section 2.2 and Appendix B, a priori error estimates have similar expressions to those of classical FE. For instance, for second-order elliptic problems the following theorem holds:

Theorem 1. *Let $\mathcal{T}_h$ be a non-degenerate discretization in elements (i.e. there is a positive constant β such that $\varrho_e/h_e \geq \beta$, for all $\Omega_e \in \mathcal{T}_h$, where h_e and ϱ_e are the diameters of Ω_e and of the sphere inscribed in Ω_e, respectively). Assuming that all boundary conditions along curved boundaries are imposed in weak form and no interior curved faces/edges are present in the mesh, the following a priori estimate holds*

$$\|u - u_h\|_{E(\Omega)} \leq K h^p |u|_{\mathcal{H}^p(\Omega)}, \tag{2.3}$$

where $\|\cdot\|_{E(\Omega)}$ is the energy norm, $u \in \mathcal{H}^{p+1}(\Omega)$ and u_h are the exact and the NEFEM solutions respectively, K is a constant depending on β, h is the mesh size, and p is the polynomial degree of interpolation.

Moreover, for p-refinement convergence the following estimate also holds,

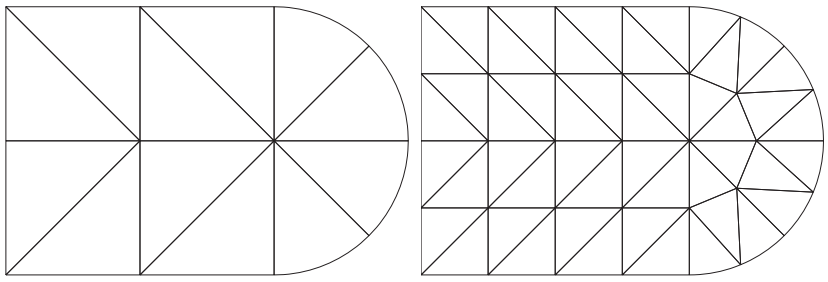
$$\|u - u_h\|_{E(\Omega)} \leq C \exp(-kN^r), \tag{2.4}$$

where C and k are positive constants, N is the number of degrees of freedom, and $r \gtrsim 1/n_{sd}$, with n_{sd} the number of spatial dimensions.

Theorem 1 assumes that essential boundary conditions are imposed in weak form, for instance with numerical fluxes in a DG context, or with Nitsche's method in a continuous formulation, see for instance Fernández-Méndez and Huerta (2004). If Dirichlet boundary conditions are imposed in strong form, an additional condition is required to keep optimal convergence rates: optimal nodal distributions on every curved element have to be considered, see Figure 2.7 (b). Theorem 1 also assumes that no interior curved faces are present in the mesh. Again, this extra hypothesis is only needed if a continuous Galerkin approximation is considered and optimal nodal distributions are required to keep optimal convergence rates. This is formally stated in the next result.

Theorem 2. *Under the assumptions of Theorem 1, the error bounds* (2.3) *and* (2.4) *hold for NEFEM in a continuous Galerkin framework, if optimal nodal distributions on every curved element along the Dirichlet boundary or with interior curved faces/edges are considered.*

The proofs of Theorems 1 and 2 are discussed in Appendix B. Several examples are shown in order to check the optimal convergence rates in a continuous and a discontinuous Galerkin framework, with different nodal distributions, see also Sevilla et al. (2008a).



(a) Mesh 1 (b) Mesh 2

Figure 2.10: Coarse meshes for h-refinement test. Nested remeshing is used for refinement.

2.5 Numerical examples

The behavior of NEFEM is illustrated using the following second-order elliptic problem:

$$\left\{\begin{array}{rcll} -\Delta u + u & = & s & \text{in } \Omega \\ u & = & u_d & \text{on } \Gamma_d \\ \nabla u \cdot \boldsymbol{n} & = & g_n & \text{on } \Gamma_n \end{array}\right. \tag{2.5}$$

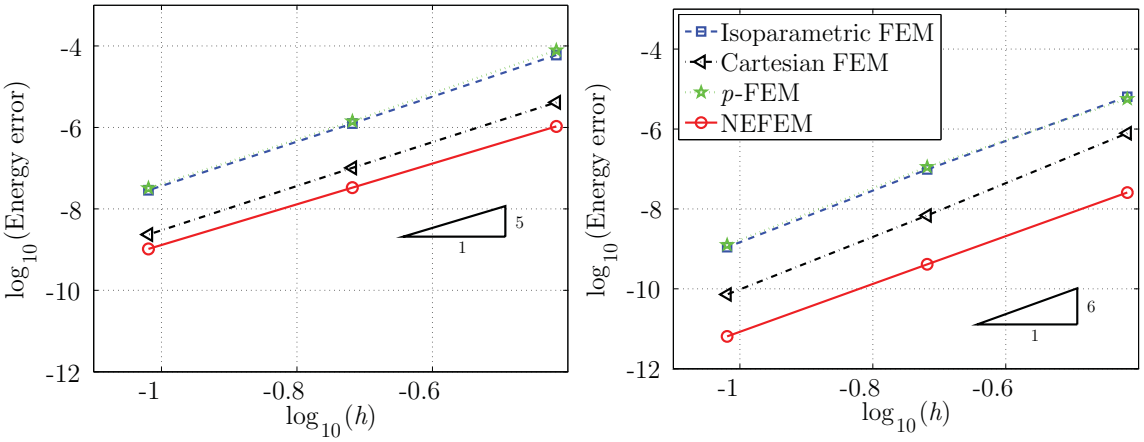
where Ω is the domain, Γ_d and Γ_n denote the Dirichlet and Neumann boundaries respectively, and $\boldsymbol{n}$ is the outward unit normal vector on $\partial\Omega = \overline{\Gamma}_d \cup \overline{\Gamma}_n$.

The 2D case is considered first. Two NEFEM meshes of the domain Ω are represented in Figure 2.10. A Dirichlet boundary condition, corresponding to the analytical solution u, is imposed in the polygonal part of the boundary Γ_d, and a Neumann boundary condition, also corresponding to the analytical normal flux, is imposed in the curved part of the boundary Γ_n. The analytical solution is $u(x,y) = x\cos(y)+y\sin(x)$, and the source term s is determined by analytical differentiation of u. The curved part of the boundary, corresponding to half of a circle, is exactly described with NEFEM using one quadratic trimmed NURBS.

Theoretical convergence rates of Theorem 1 are checked in Appendix B. Here NEFEM performance is compared with isoparametric FEM, cartesian FEM and p-FEM. Main differences between these methodologies are recalled here, see Appendix C for a review and further comparison of these techniques.

Classical isoparametric FEs consider a nodal interpolation of the solution with local coordinates. An isoparametric mapping is defined to relate local and cartesian (physical) coordinates. Thus, there are two focus of error in the isoparametric FEM. First, the isoparametric mapping introduces geometric errors, due to the approximation of the physical element Ω_e by the computational element Ω_e^h. In fact, the boundary of the computational domain $\partial\Omega_h$ is a piecewise polynomial approximation of the exact boundary $\partial\Omega$. On the other hand, for high-order approximations on curved elements, the definition of the polynomial interpolation in local coordinates implies a loss of consistency: a polynomial interpolation of degree $p > 1$ in local coordinates does not correspond to a polynomial interpolation of degree p in cartesian coordinates. An alternative to ensure consistency of the approximation is the so-called cartesian FEM. In this approach, the polynomial basis for the approximation of the solution is defined with cartesian coordinates. Nevertheless, the isoparametric transformation and the computational element Ω_e^h, are still considered for integration purposes. Thus, although cartesian FEM gets rid of the consistency lack of isoparametric FEM, it still maintains the geometric error. This is not the case for p-FEM and NEFEM. p-FEM considers the exact boundary representation and the approximation is defined in a reference element. Thus, the difference between isoparametric FEs and p-FEM is only due to geometry, and the difference between p-FEM and NEFEM is only due to the definition of the high-order polynomial basis with local coordinates, not in the physical space as done by NEFEM,

Figure 2.11 shows a comparison of the convergence under h-refinement for several FE techniques. Energy error is depicted for a polynomial approximation of degree $p = 5$ and $p = 6$. The optimal rate of h-convergence is exhibited by every FE technique considered, but some differences in accuracy are observed. In this example the use of a cartesian approximation (cartesian FEs and NEFEM) provides more accurate results than defining the approximation with local coordinates. In all the computations, the use of NEFEM provides the most accurate results due to the cartesian approximation combined with an exact boundary representation. With $p = 6$, NEFEM is one order of magnitude more accurate than cartesian FEs and two orders of magnitude more



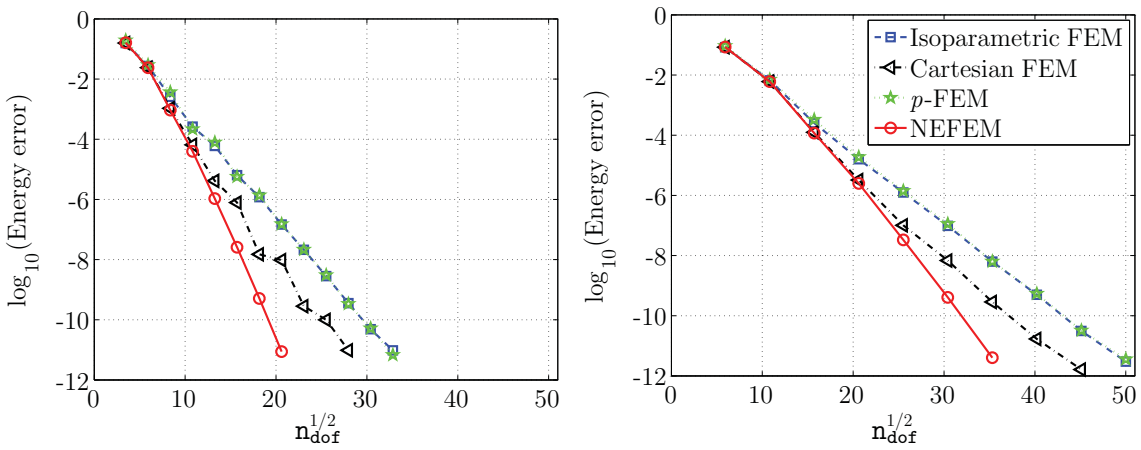
(a) $p = 5$ (b) $p = 6$

Figure 2.11: Second-order elliptic problem: h-convergence in the energy norm

accurate than isoparametric FEs and p-FEM. Cartesian FEs provide more accurate results than isoparametric FEs and p-FEM. In this example, p-FEM does not represent an advantage with respect to isoparametric FEs. The error due to the geometric approximation of the boundary is lower than the error introduced by the definition of the polynomial basis in local coordinates.

Next, convergence under p-refinement is explored and compared. Figure 2.12 represents the evolution of the energy error as a function of the square root of the number of degrees of freedom ($\mathtt{n_{dof}}$). The polynomial degree of the approximation is uniformly increased starting with $p = 1$ and for the discretizations shown in Figure 2.10. As the order of the polynomial approximation is increased, NEFEM offers the best performance. In fact, the desired error is attained with the minimum $\mathtt{n_{dof}}$. Figure 2.12 shows that, for a given accuracy and the coarsest mesh in Figure 2.10, NEFEM allows to reduce drastically the $\mathtt{n_{dof}}$. In particular, a reduction of 40% compared to cartesian FEM and up to 50% compared to isoparametric FEM or p-FEM.

Finally, the influence of the number of integration points on the accuracy is studied. The coarsest mesh in Figure 2.10 with a polynomial approximation of degree $p = 6$ and $p = 8$ is used. To study quadrature accuracy, Figure 2.13 shows the evolution of the energy error versus the number of Gauss integration points for every



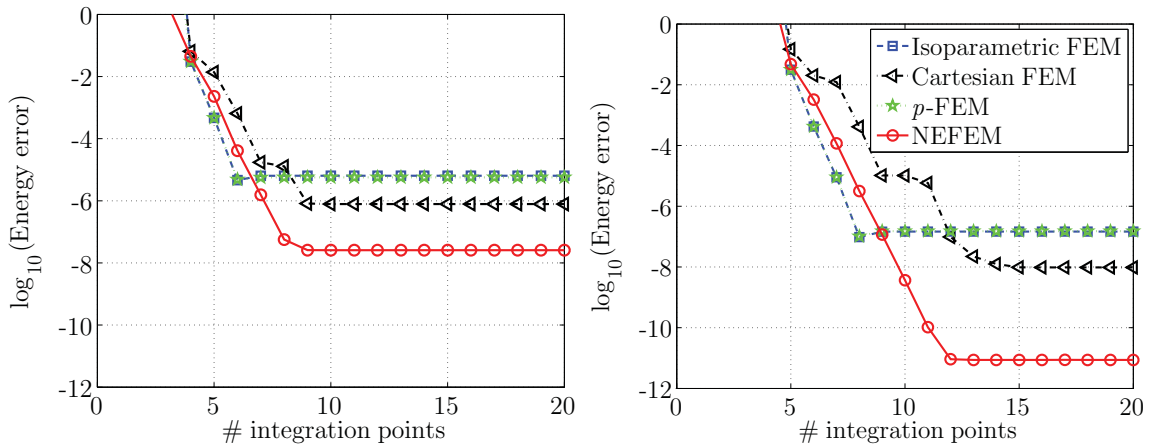
(a) Mesh 1 (b) Mesh 2

Figure 2.12: Second-order elliptic problem: p-convergence in the energy norm for the discretizations in Figure 2.10

curved boundary edge. When the polynomial basis is defined in cartesian coordinates (cartesian FEM and NEFEM), numerical integration requires more integration points to reach its maximum accuracy, compared to the other methods. For a given degree of interpolation, NEFEM is able to reach the same accuracy of isoparametric FE with only one extra integration point. Moreover, with three or four integration points more than isoparametric FEM NEFEM reaches its maximum accuracy. For a degree of interpolation $p = 8$, NEFEM is four orders of magnitude more precise than isoparametric FEM and p-FEM, and three orders of magnitude more precise than cartesian FEM.

The second-order elliptic problem (2.5) is next considered in 3D, where Ω is a sphere of unit radius. The analytical solution is $u(x, y) = x\cos(y) + y\sin(z) + z\cos(x)$, and the source term s is again determined by analytical differentiation of u. A Neumann boundary condition, corresponding to the analytical normal flux is imposed in $\partial\Omega$. A coarse mesh with only eight curved tetrahedral elements is considered for the simulations, see Figure 2.14, and high-order approximations are introduced to properly capture the solution.

Figure 2.15 shows FEM and NEFEM solutions with quadratic and cubic approxi-



(a) $p = 6$ (b) $p = 8$

Figure 2.13: Second-order elliptic problem: energy norm of the error as the number of integration points per curved boundary edge is increased, for the coarsest mesh in Figure 2.10

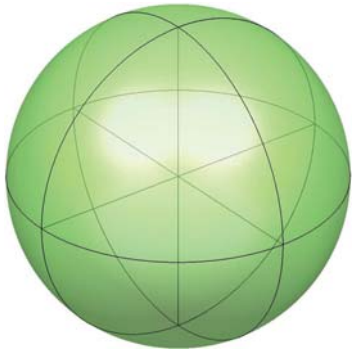
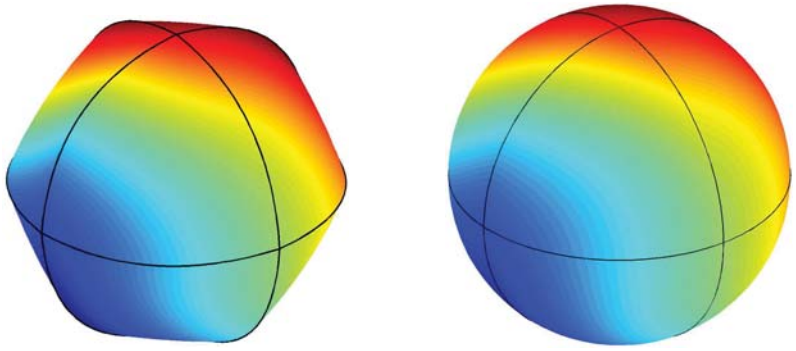
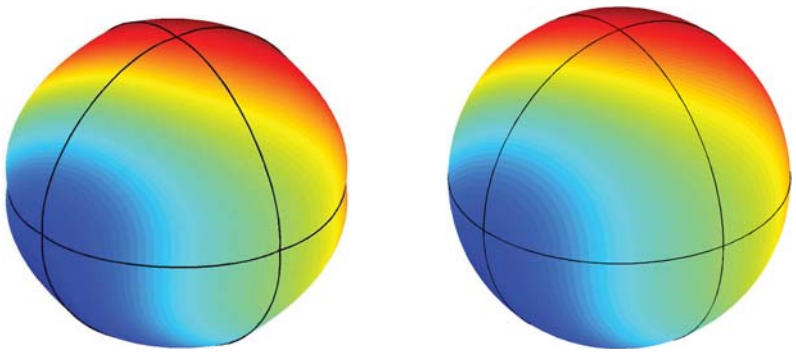


Figure 2.14: Coarse mesh of the sphere with eight curved tetrahedrons

mation. The piecewise polynomial approximation of the curved boundary introduced by the isoparametric mapping is clearly observed. With quadratic FE, the maximum difference between the exact and the approximated boundary is 0.1037. For cubic approximation, the geometric error is still important, 0.0268. Moreover, the piecewise polynomial approximation of the boundary induces a loss of regularity. The exact boundary $\partial\Omega$ is a $\mathcal{C}^\infty$ surface, whereas the approximation given by isoparametric or cartesin FEs, namely $\partial\Omega^h$, is only $\mathcal{C}^0$ along boundary edges, see Figures 2.15 (a) and (c). In NEFEM, the boundary is exactly described with one quadratic singular



(a) FEM p=2 (b) NEFEM p=2



(c) FEM p=3 (d) NEFEM p=3

Figure 2.15: Second-order elliptic problem:: surface plot of FEM and NEFEM solutions using quadratic and cubic approximations

NURBS, with no dependence on the spatial discretization (i.e. the polynomial degree of the approximation), as represented in Figures 2.15 (b) and (d).

Figure 2.16 shows a p-convergence comparison when the polynomial order of the approximation is uniformly increased starting with $p = 2$ and for the mesh shown in Figure 2.14. Errors in the maximum and in the energy norm are represented as a function of the cube root of $\mathtt{n_{dof}}$. For NEFEM, the expected (exponential) convergence for a problem with a smooth solution is obtained, whereas for methods with an approximate boundary representation, a much slower convergence is obtained. Note that cartesian and isoparametric FEs offer the same performance if the error is measured in the maximum norm. However, when the error is measured in the energy norm, cartesian FEs performs better. The definition of the polynomial basis

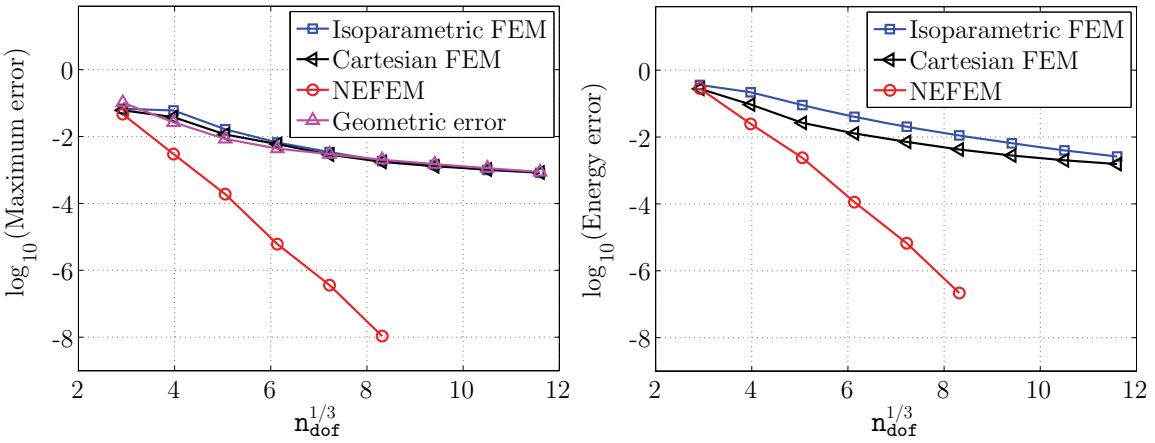


Figure 2.16: p-convergence comparison for the second-order elliptic problem. The polynomial degree of the approximation is uniformly increased from $p = 2$ and the error is measured (a) in the maximum norm and (b) in the energy norm

in cartesian coordinates offers a better approximation of the derivatives, compared to isoparametric FEs, as in the previous 2D examples. Figure 2.16 (a) also depicts the maximum geometric error (measured as the maximum distance between the true boundary $\partial\Omega$ and the piecewise polynomial approximation of the boundary given by $\partial\Omega^h$), revealing that if an approximated boundary representation is considered (isoparametric FEM or cartesian FEM), the error of the solution is controlled by the error on the geometry.

This example reveals the importance of the geometrical model in FE simulations and critical conclusions are derived. In Szabó and Babuška (1991), p-FEM with an exact boundary description is compared with high-order subparametric elements (with a quadratic approximation of the boundary). Two dimensional examples confirm the expected exponential convergence of p-FEM, whereas the subparametric approach leads to a suboptimal rate of convergence. The 3D example shown in this section shows a more dramatically situation because NEFEM is compared with high-order isoparametric elements, and the conclusions are the same as those obtained by Szabó and Babuška (1991). Therefore, this example demonstrate that a high-order approximation of the geometry is not always sufficient to achieve the maximum accuracy.

Chapter 3

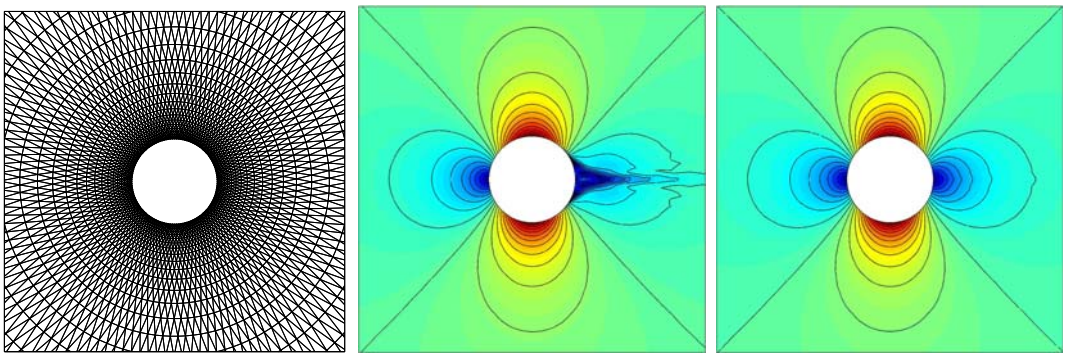
Discontinuous Galerkin NEFEM for electromagnetic scattering

Electromagnetic scattering problems model the interaction between an incident electromagnetic wave and a general scatterer. This chapter presents the application of NEFEM to time-domain electromagnetic scattering problems using a DG formulation.

Maxwell's equations, relating the electric and magnetic scattered fields, $\boldsymbol{E} = (E_1, E_2, E_3)^T$ and $\boldsymbol{H} = (H_1, H_2, H_3)^T$ respectively, are described in detail in Appendix D. The decoupling into the so-called transverse electric (TE) and transverse magnetic (TM) modes is recalled. The DG formulation is described, with particular emphasis on the treatment of boundary conditions. The radar cross section (RCS), which is a usual quantity of interest in electromagnetic scattering simulations, is also recalled in Appendix D, and it is used in the numerical computations to measure their accuracy.

Moreover, a key issue in the numerical solution of electromagnetic scattering problems is the use of a mechanism to perform the absorption of outgoing waves. The numerical examples presented in this chapter consider the non-linear PML proposed by Abarbanel et al. (2006). In Appendix E, this PML and the original PML proposed by Berenger (1994) are reviewed, optimized and compared, in a high-order DG framework.

In this chapter, several numerical examples are shown to illustrate the possibilities and benefits of NEFEM. Some validation examples are used to test the performance

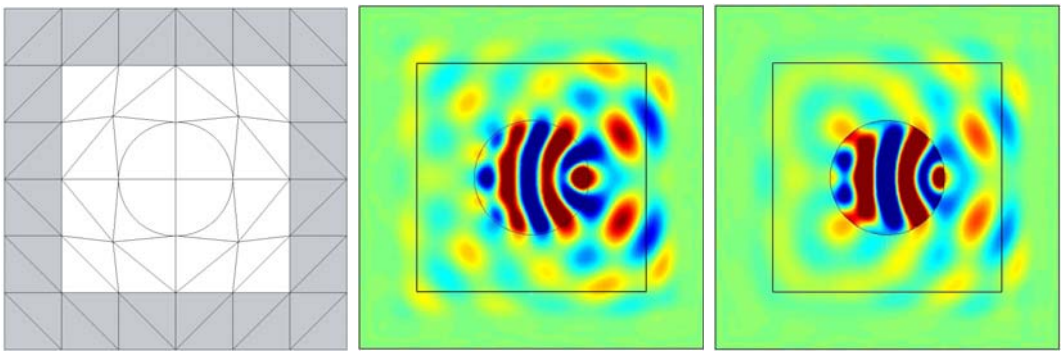


(a) 128 × 32 mesh (b) FEM (c) NEFEM

Figure 3.1: Inviscid subsonic flow around a circle: FEM and NEFEM Mach number distributions and isolines with linear elements

of NEFEM, including domains with material interfaces and comparison with other techniques used by the CEM community. A comparison of NEFEM versus several curved FEs is presented, showing the potential of the proposed formulation. The behavior and benefits of NEFEM in the presence of complex scatterers with small geometric features is also explored. Finally, more challenging problems are considered, including complex geometries and higher frequencies.

This chapter focuses on electromagnetic scattering problems, nevertheless, it is worth mentioning that NEFEM exhibits important advantages in other areas, such as the DG solution of compressible flow problems, see Appendix F and Sevilla et al. (2008b). To illustrate the performance of NEFEM in the numerical solution of Euler equations, the subsonic inviscid flow around a circle is considered. Figure 3.1 (a) shows a detailed view of a mesh with 128 × 32 nodes. The Mach number distribution and isolines computed with isoparametric FEs are represented in Figure 3.1 (b). As reported by Bassi and Rebay (1997), using linear isoparametric elements, a spurious entropy production behind the circle prevents the convergence to the steady state solution, even if the mesh is drastically refined near the curved boundary. Whereas NEFEM is able to converge to the correct solution even if a linear approximation of the solution is considered, see the Mach number distribution and isolines obtained with NEFEM and linear approximation in Figure 3.1 (c).



(a) Mesh (b) H_3 field (TE mode) (c) E_3 field (TM mode)

Figure 3.2: Scattering by a dielectric cylinder of diameter 2λ: computational mesh with a λ thick PML and transverse scattered fields for a NEFEM solution with $p = 8$

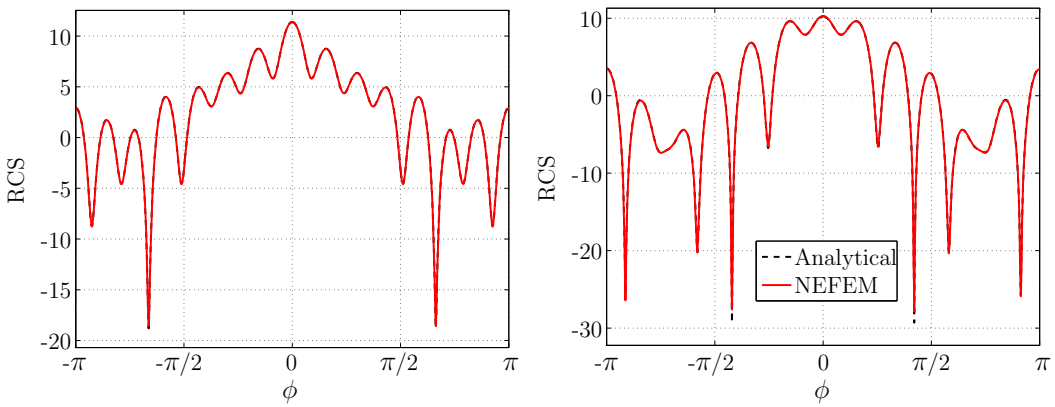
3.1 Validation examples

This section presents three examples used to validate the DG NEFEM solution of electromagnetic scattering problems in 2D and 3D.

3.1.1 Dielectric circular cylinder

The first example considers the TE and TM modes for the scattering of an incident plane wave travelling in the x^+ direction by a dielectric cylinder surrounded by free-space ($\varepsilon = \mu = 1$). The diameter of the cylinder is 2λ and the material parameters of the dielectric media are $\varepsilon_d = 2.56$ and $\mu_d = 1$, see Appendix D and Morgan et al. (2000) for the details about the simulation of electromagnetic scattering problems in a piecewise homogeneous media. The analytical solution for this problem can be found in Harrington (1961) or Balanis (1989).

Figure 3.2 (a) shows the computational mesh used in the computations. It has only four elements to discretize the dielectric media, 28 elements in the free-space region and a λ thick PML with 40 elements. The interface between free-space and the dielectric cylinder is exactly described using one quadratic NURBS curve. Figures 3.2 (b) and (c) show the TE and TM transverse scattered field for a NEFEM solution with $p = 8$. Note that the electric field is discontinuous across the material interface



(a) TE mode (b) TM mode

Figure 3.3: Scattering by a dielectric cylinder of diameter 2λ: RCS for a NEFEM solution with $p = 8$

between the free-space and the dielectric media. Owing to the discontinuity of the electric permittivity ε, only the tangential component of the electric field is continuous across such interfaces, see Appendix D. The DG formulation considered here allows a natural representation of the discontinuous solution.

Figure 3.3 compares the computed and analytical RCS distributions for the TE and TM modes. The NEFEM solution and the analytical solution overlap. Moreover, Figure 3.4 shows a p-convergence study for the TE and TM modes. The relative RCS error in $\mathcal{L}^2(-\pi, \pi)$ norm is represented as a function of the square root of the number of degrees of freedom ($\mathtt{n_{dof}}$). First a *preasymptotic* regime is observed up to $p = 5$, where the error oscillates as the degree of approximation is increased due to dispersion errors. Increasing the degree of the approximation up to $p = 7$, a *transition* regime is observed, with an algebraic convergence rate. Finally, for $p > 7$ the *resolved* regime is observed, with the expected (exponential) convergence rate for a problem with smooth solution, see Ainsworth (2004) for further details on the effect of dispersive errors in high-order DG methods.

It is worth noting that using low order methods, 20 nodes per wavelength are usually considered near the scatterer, see, among others, Morgan et al. (2000). The exact boundary representation considered by NEFEM allows to use coarse meshes and

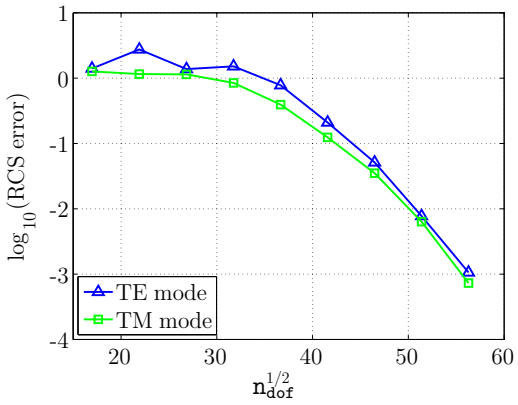


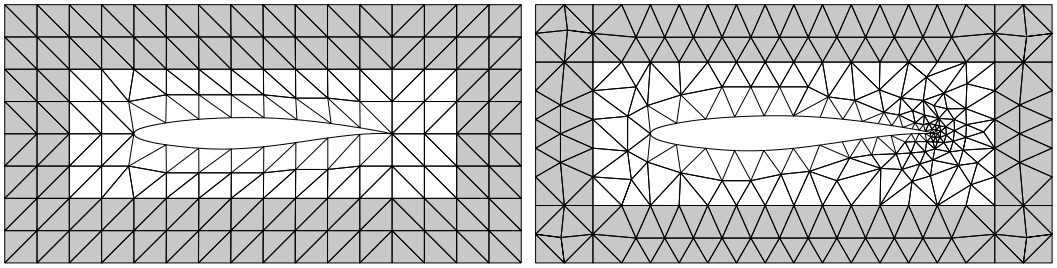
Figure 3.4: Scattering by a dielectric cylinder of diameter 2λ: p-convergence of the RCS error in $\mathcal{L}^2(-\pi, \pi)$ norm

achieve convergence by increasing the degree of the approximation. In this example, a relative RCS error of 10^{-2} is provided by a degree of approximation $p = 8$, that is 9 nodes per wavelength, showing the potential of NEFEM.

3.1.2 PEC RAE2822 airfoil

The following example considers the scattering by a PEC RAE2822 airfoil of chord length 5λ. The incident wave travels with an angle of $\pi/4$ with respect to the x axis. The RAE2822 is a non-symmetric airfoil with analytical expression (Selig, 2008) that can not be exactly described with a NURBS curve. As usual in the context of airfoil shape optimization, an approximation using B-splines is considered here.

Since no analytical solution is available, two comparisons are presented in order to validate the results. The first comparison involves the computation with NEFEM in a coarse mesh with uniform element size and 256 elements, see Figure 3.5 (a), and in a mesh refined towards the trailing edge with 398 elements, see Figure 3.5 (b). The refined mesh is designed to isolate the effect of the singularity exhibited by the scattered field at the trailing edge. The TE scattered fields are depicted in Figure 3.6 for a NEFEM solution with $p = 8$ in both meshes. To achieve the time-harmonic steady state, the solution is advanced in time until the difference in the scattering width ($10^{RCS/10}$, see Appendix D) between two consecutive cycles is lower than 10^{-5}

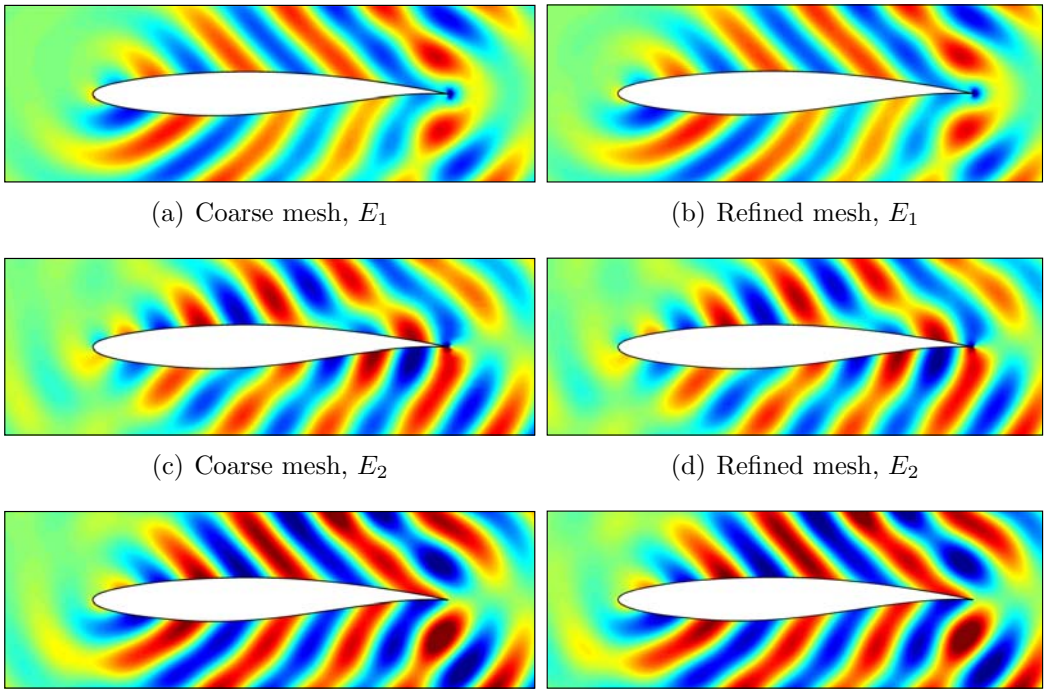


(a) Coarse mesh (b) Refined mesh towards the singularity

Figure 3.5: Computational meshes for the scattering by a PEC RAE2822 airfoil of chord length 5λ

in the $\mathcal{L}^\infty(-\pi,\pi)$ norm. If the RCS is used in the stop criteria slower convergence is usually obtained due to the singularities introduced by the logarithmic scale, see Appendix D. The computation in the coarse mesh requires 26 510 fourth order Runge-Kutta time steps, whereas the computation in the refined mesh takes 188 622 time steps. Furthermore, the computational cost per time step is lower in the coarse mesh due to the lower number of elements.

A detailed view of the TE scattered fields in the free-space region is depicted in Figure 3.6. In these plots, the scattered field distributions are indistinguishable, even in the vicinity of the singularity. The absolute error of the fields over the lower and upper parts of the airfoil is represented in Figure 3.7, showing the good performance of NEFEM in the coarse mesh. The maximum discrepancy is observed at the trailing edge, $x = 0.5$, as expected due to the singularity of the fields. But, it is important to note that this maximum difference is extremely localized. More precisely, an error below 5×10^{-2} is observed in 99.2% of the airfoil profile. Lower errors are obtained for the E_1 field, but it is only due to the smoother variations of this scattered field component. Figure 3.8 (a) compares the TE RCS distribution obtained in both meshes. An excellent agreement between both RCS patterns is observed, with a difference between the solution in the coarse mesh and the finer one of 4.8×10^{-2} in the $\mathcal{L}^2(-\pi,\pi)$ norm. Thus, the discrepancy of the fields in the singular point does not translate in any relevant difference in the RCS distribution. Therefore, extremely refined meshes towards singular points are not always mandatory to obtain accurate RCS distributions.

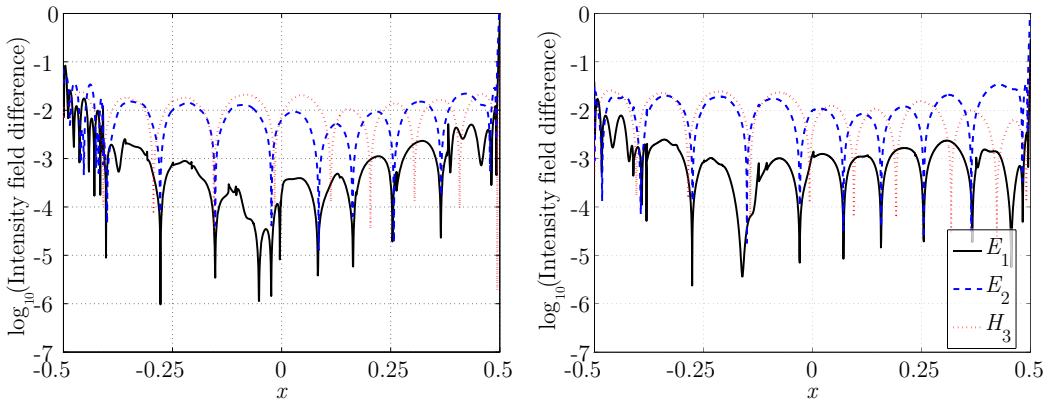


(a) Coarse mesh, E_1 (b) Refined mesh, E_1

(c) Coarse mesh, E_2 (d) Refined mesh, E_2

(e) Coarse mesh H_3 (f) Refined mesh, H_3

Figure 3.6: Scattering by a PEC RAE2822 airfoil of chord length 5λ: detail of the scattered fields in the discretizations shown in Figure 3.5



(a) (b)

Figure 3.7: Scattering by a PEC RAE2822 airfoil of chord length 5λ: difference of the intensity fields in the (a) upper and (b) lower parts of the airfoil

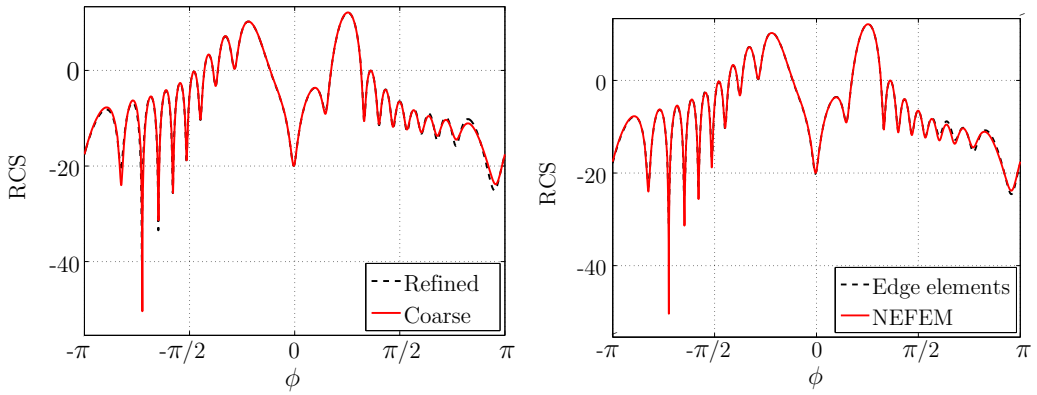
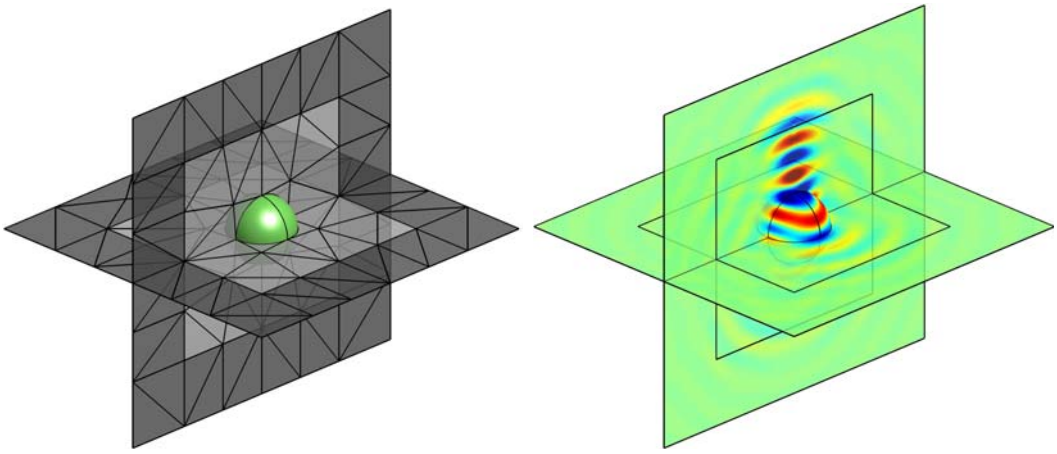


Figure 3.8: Scattering by a PEC RAE2822 airfoil of chord length 5λ: RCS comparison (a) NEFEM in the coarse and fine meshes, and (b) NEFEM in the coarse mesh and high-order edge elements

Airfoils are widely used for testing scattering codes, see, among others, Bonnet and Poupaud (1997), Jiang (1998) and Ledger (2001). But it is worth noting that, in the mentioned works, not only h-refinement towards the singularity is introduced, but also the mesh is refined towards the leading edge, $x = -0.5$ to provide an accurate geometry description. Figure 3.8 (b) compares the TE RCS computed with NEFEM in the coarse mesh of Figure 3.5 (a) with the solution obtained by using high-order edge elements (Ledger, 2001) with $p = 8$. The mesh used in the computation with edge elements has 1 901 triangles in the free-space region, and refinement towards the leading and trailing edges of the airfoil is performed to accurately capture the geometry and to isolate the effect of the singularity. Again, an excellent agreement in the RCS is observed, with a difference of 4.4×10^{-2} in the $\mathcal{L}^2(-\pi, \pi)$ norm. Note that the coarse mesh used with NEFEM has only 96 triangular elements in the free-space region, and almost identical RCS distributions are obtained. This comparison corroborates that reliable results are obtained with NEFEM using coarse meshes and high-order approximations, and therefore, with important savings in computational cost and memory requirements.



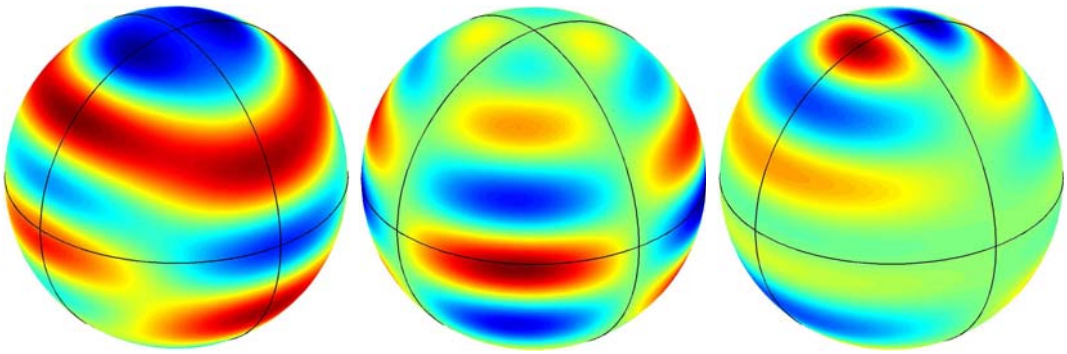
(a) (b)

Figure 3.9: Scattering by a PEC sphere of diameter 2λ: (a) two cuts of a coarse mesh with a 2λ thick PML, and (b) E_1 field for a NEFEM solution with $p = 9$

3.1.3 PEC Sphere

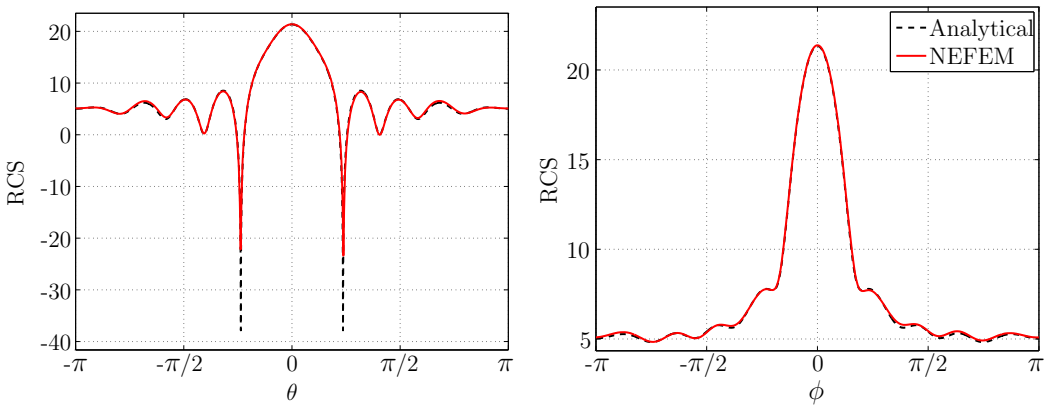
An incident plane wave travelling in the z^+ direction scattered by a sphere of diameter 2λ is considered next. The analytical solution for this problem can be found in Harrington (1961) or Balanis (1989).

The sphere is exactly described with a quadratic singular NURBS surface, and a coarse mesh with only eight elements for the discretization of the curved boundary is considered, see two cuts of the volume mesh and the surface mesh of the sphere in Figure 3.9 (a). The mesh has 1 271 elements with planar faces and 32 curved elements (8 elements with a face on the NURBS boundary and 24 elements with an edge on the NURBS boundary). The scattered E_1 field computed with NEFEM and a polynomial approximation of degree $p = 9$ is represented in Figure 3.9 (b), showing the field on the sphere surface and illustrating the absorption of the outgoing waves in the PML. A detailed view of the scattered electric field over the surface of the sphere is represented in Figure 3.10, showing the complex behavior of the solution captured with only eight tetrahedral elements with a face on the sphere. Finally, Figure 3.11 shows a comparison between the computed and the exact RCS. An excellent agreement is obtained for both horizontal and vertical polarizations. More precisely, the relative



(a) E_1 (b) E_2 (c) E_3

Figure 3.10: Scattering by a PEC sphere of diameter 2λ: detail of the scattered electric field over the sphere for a NEFEM solution with $p = 9$

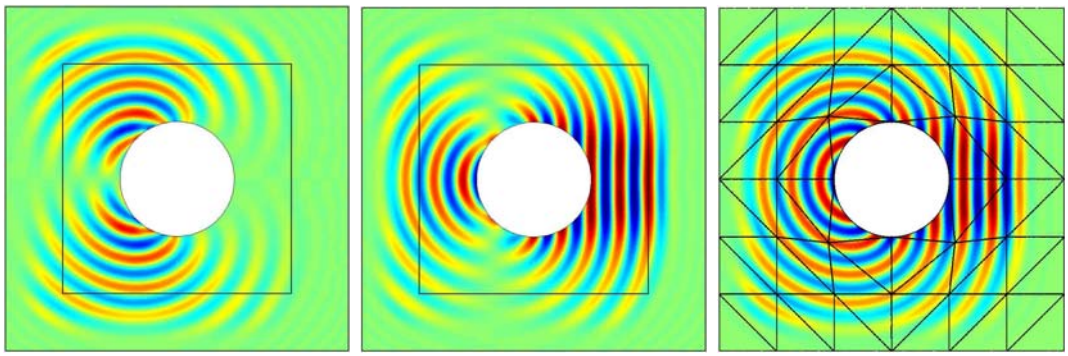


(a) Vertical polarization (b) Horizontal polarization

Figure 3.11: Scattering by a PEC sphere of diameter 2λ: RCS for a NEFEM solution with $p = 9$

RCS error in $\mathcal{L}^2([-\pi, \pi])$ norm is 7.2×10^{-2} for the vertical polarization and 9.1×10^{-3} for the horizontal polarization. Note that the RCS for vertical polarization exhibits a singularity due to the logarithmic scale. In fact, if the error is measured in the scattering width, similar accuracy is obtained for vertical and horizontal polarizations, namely 7.6×10^{-3} and 7.2×10^{-3} respectively.

Note that, again, the solution is accurately captured with one element per wavelength and a degree of approximation $p = 9$, that is, with 10 nodes per wavelength. Compared to other methods, this represents an important save in memory and com-



(a) H_1 (b) H_2 (c) E_3

Figure 3.12: Scattering by a PEC cylinder of diameter 4λ: scattered fields for a NEFEM solution with $p = 10$

putational cost. For instance, Vinh et al. (1997) use 22 nodes per wavelength using low-order FDs, Hachemi et al. (2004) use 20 nodes per wavelength with linear FEs and Cioni et al. (1993) uses 20 nodes per wavelength with FVs.

3.2 NEFEM performance comparison

In this section, the performance of NEFEM is compared with several FE techniques for the treatment of curved boundaries: isoparametric FEM, cartesian FEM and p-FEM. These techniques are recalled and compared with NEFEM from a theoretical point of view in Appendix C.

3.2.1 PEC circular cylinder

The first example consists on a planar TM wave travelling in the x^+ direction and scattered by a PEC circular cylinder of diameter 4λ. The scattered fields for a NEFEM approximation of degree $p = 10$ are represented in Figure 3.12. The computational mesh is also displayed, with only four curved elements to discretize the circle. Note that a 2λ thick PML is introduced, which is enough to ensure that the accuracy comparison is not affected by the PML, see Appendix E.

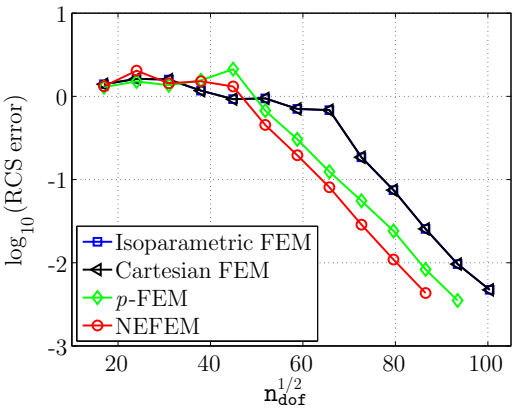


Figure 3.13: Scattering by a PEC cylinder of diameter 4λ: p-convergence comparison of several curved FEs

The RCS error evolution for increasing p is depicted in Figure 3.13. For the same discretization (i.e. same degree of interpolation), NEFEM results are more accurate than isoparametric or cartesian FE, with an approximate boundary description, and also more accurate than p-FEM, with an exact boundary representation. For instance, NEFEM with $p = 10$ produces a RCS error in $\mathcal{L}^2([-\pi, \pi])$ norm of about 10^{-2}, whereas isoparametric or cartesian FE require $p = 12$ to achieve a comparable accuracy, and p-FEM requires $p = 11$. Thus, NEFEM is able to reach the desired accuracy with a reduction of about 30% compared to isoparametric or cartesian FEs, and of 15% compared to p-FEM (also with an exact boundary representation). This difference in number of degrees of freedom implies important differences in computational cost. NEFEM computation requires 2 585 time steps to reach the steady state, whereas isoparametric and cartesian FEs employ 3 692 time steps and p-FEM requires 3 114 time steps. In addition, each time step of the NEFEM computation requires less computational cost due to the lower p needed to achieve the desired accuracy.

The difference between isoparametric FEs and cartesian FEs are indistinguishable, showing that a cartesian approximation of the solution does not offer any advantage if an approximated boundary representation is considered. The difference between isoparametric FEs and p-FEM is only due to geometric errors, and relevant differences in accuracy are observed. Recall that the exact boundary representation is

crucial in this example because and due the weak imposition of the PEC boundary condition using the exact outward unit normal. Finally, NEFEM also considers the exact boundary representation and outperforms p-FEM, showing that the cartesian approximation combined with an exact boundary representation, i.e. NEFEM, provides the maximum accuracy for a given spatial discretization. Finally, note that with an approximate boundary representation the exponential convergence is exhibited for $p > 8$ whereas with an exact boundary representation the exponential convergence is achieved for $p > 5$.

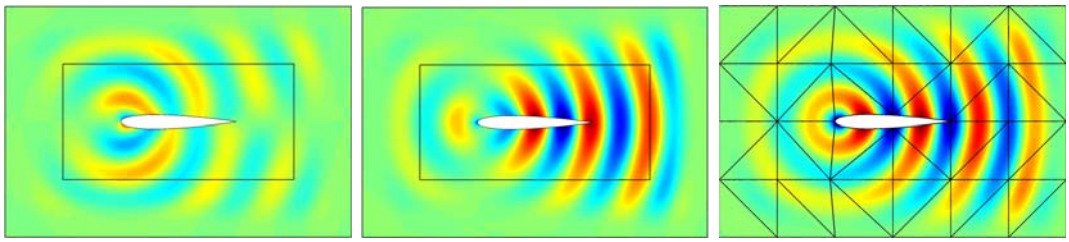
To conclude, it is worth remarking that only one element per two wavelengths is considered in this example and a RCS error of order 10^{-3} is obtained with $p = 11$, that is, using 6 nodes per wavelength. Thus, the exact geometry considered in NEFEM combined with the cartesian approximation allows to computed accurate solutions with the minimum number of degrees of freedom, compared to other curved FEs and other techniques used by the CEM community.

3.2.2 PEC NACA0012 airfoil

A planar TE wave travelling in the x^+ direction and scattered by the NACA0012 airfoil of chord length 2λ is considered. The NACA0012 is a symmetric airfoil with analytical expression (Ladson et al., 1996) that can not be exactly described with a NURBS curve. As usual in the context of airfoil shape optimization, an approximation of the airfoil using B-splines is considered here.

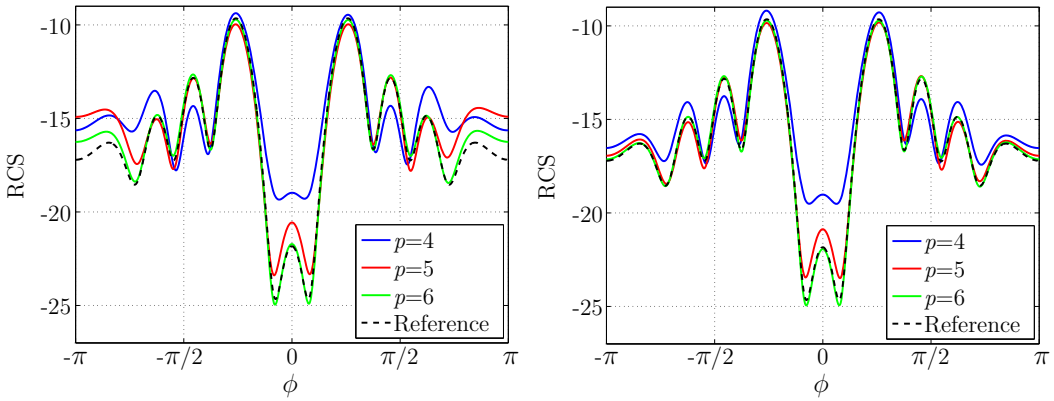
Figure 3.14 shows the computational mesh with a λ thick PML and the scattered fields for a NEFEM solution with $p = 6$. Figure 3.15 illustrates the convergence of the solution for increasing p, with isoparametric FEs and NEFEM. As no analytical solution is available, a reference solution is computed in a fine mesh with high-order approximation and a thicker PML.

For isoparametric FEs, the RCS error in $\mathcal{L}^2([-\pi, \pi])$ norm decreases as p increases, but it is important to remark that the RCS error in the $\mathcal{L}^\infty([-\pi, \pi])$ norm behaves different. In particular, the RCS error for angles near $-\pi$ or π is higher with $p = 5$



(a) E_1 (b) E_2 (c) H_3

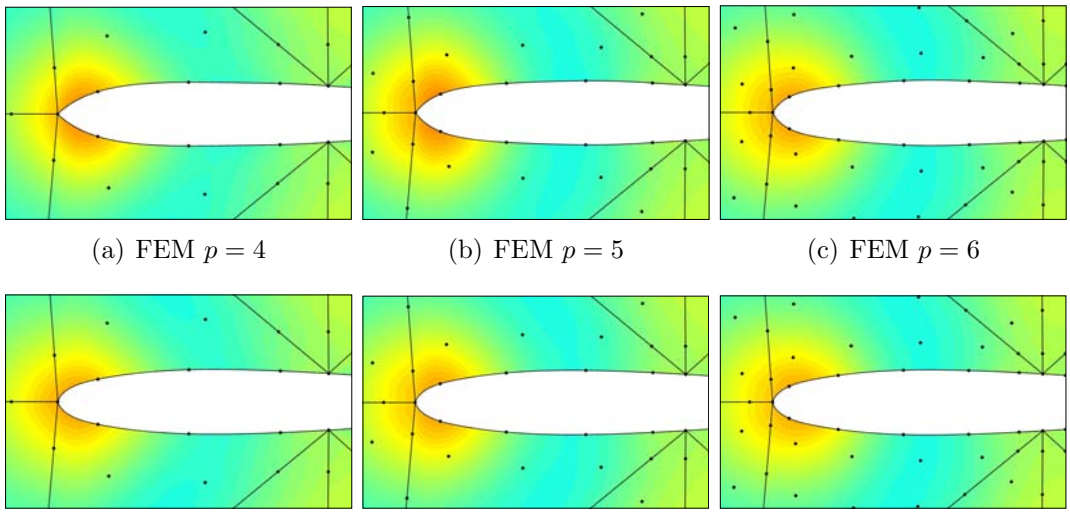
Figure 3.14: Scattering by a PEC NACA0012 airfoil of chord length 2λ: scattered fields for a NEFEM solution with $p = 6$



(a) FEM (b) NEFEM

Figure 3.15: Scattering by a PEC NACA0012 airfoil of chord length 2λ: TE RCS comparison as p increases

than using $p = 4$, showing important discrepancies with respect to the reference solution. Recall that, for isoparametric FEs, as p increases not only the solution is represented with higher degree, but also the geometry. Thus, slightly different profiles are considered for each p. Moreover, the approximated boundary is only $\mathcal{C}^0$ on the boundary nodes. In particular a discontinuity of the profile in the leading edge is clearly observable using coarse meshes and high-order isoparametric FEs, see Figures 3.16 (a), (b) and (c). With NEFEM, the exact boundary representation is considered with no dependence on the spatial discretization (i.e. the degree of the polynomial approximation), see Figures 3.16 (d), (e) and (f). Consequently, with NEFEM ,the RCS error is uniformly reduced for all viewing angles as the degree of



(a) FEM $p = 4$ (b) FEM $p = 5$ (c) FEM $p = 6$

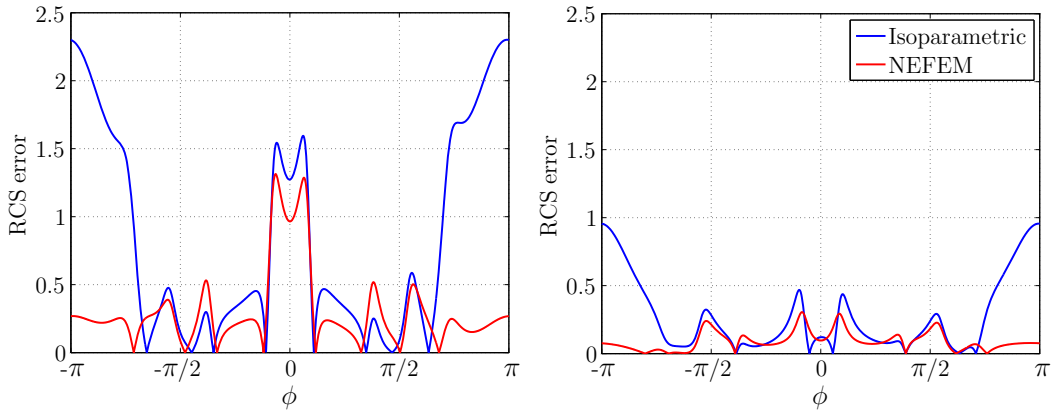
(d) NEFEM $p = 4$ (e) NEFEM $p = 5$ (f) NEFEM $p = 6$

Figure 3.16: Scattering by a PEC NACA0012 airfoil of chord length 2λ: detailed view near the leading edge for increasing p

the approximation is increased, see Figure 3.15.

Figure 3.17 compares the RCS error distribution with isoparametric FEs and NEFEM, for a degree of approximation $p = 5$ and $p = 6$. Note that the maximum error with isoparametric FEs is observed at viewing angles corresponding to the leading edge ($\phi = -\pi$ and $\phi = \pi$ angles), whereas for NEFEM the maximum error is obtained near the singularity ($\phi = 0$).

This example illustrate the sensitivity of the RCS to poor geometric representations. Isoparametric approximations are not sufficient when coarse meshes and high-order approximations are considered. Geometric errors may lead to important discrepancies in the scattered field, and therefore, in the RCS. Thus, h-refinement is usually performed at the leading edge of airfoils to provide an accurate representation of the geometry. With NEFEM, the exact boundary representation allows to mesh the domain with no dependence on the geometrical complexity. Using only one element per wavelength and $p = 5$, i.e. 6 nodes per wavelength, an accurate solution is obtained, without performing h-refinement. The maximum error in a NEFEM computations is observed where the solution is complex, not where the geometry is



(a) $p = 5$ (b) $p = 6$

Figure 3.17: Scattering by a PEC NACA0012 airfoil of chord length 2λ: RCS error comparison for isoparametric FEM and NEFEM

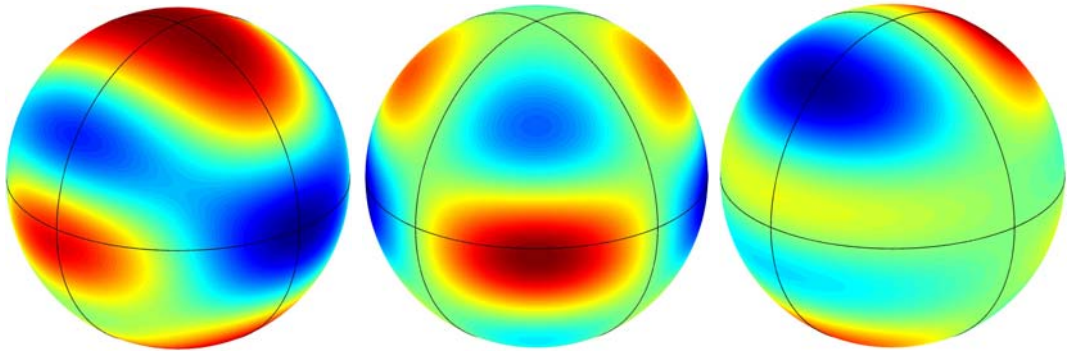
complex.

3.2.3 PEC sphere

The last example in this section considers an incident plane wave travelling in the z^+ direction and scattered by a PEC sphere of diameter λ.

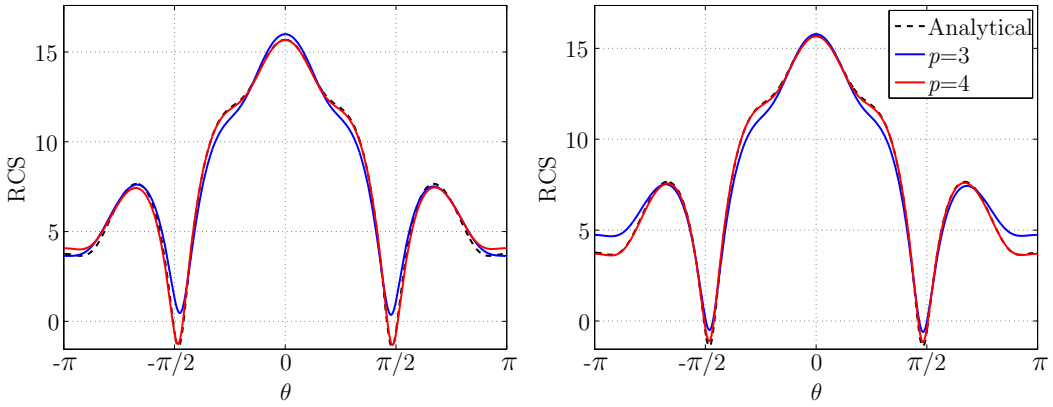
The coarse mesh represented in Figure 3.9 (a) is considered, and high-order approximations are introduced to properly capture the solution. The scattered electric field computed with NEFEM and a polynomial approximation of degree $p = 5$ is represented over the surface of the sphere in Figure 3.18.

Figures 3.19 and 3.20 compare the RCS computed with degree $p = 3$ and $p = 4$ with the analytical solution, for vertical and horizontal polarization respectively. For cartesian FEs, the RCS error is not reduced for all viewing angles as p increases. In particular, the RCS near viewing angles $-\pi$ and π is more accurate with $p = 3$ than using $p = 4$. Again, the approximate boundary representation has a critical influence in the scattered fields, and therefore, in the RCS. NEFEM exhibits the same robustness than in the previous examples. The error is decreased for all viewing angles as the degree of the approximation is increased. A perfect match between analytical



(a) E_1 (b) E_2 (c) E_3

Figure 3.18: Scattering by a PEC sphere of diameter λ: scattered electric field computed with NEFEM and a degree of approximation $p = 5$

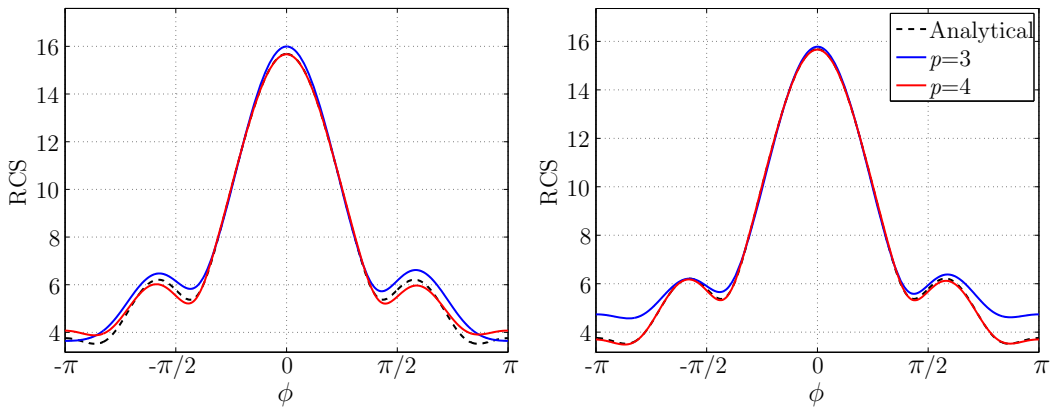


(a) Cartesian FEM (b) NEFEM

Figure 3.19: Scattering by a PEC sphere of diameter λ: RCS comparison for increasing p and for the vertical polarization

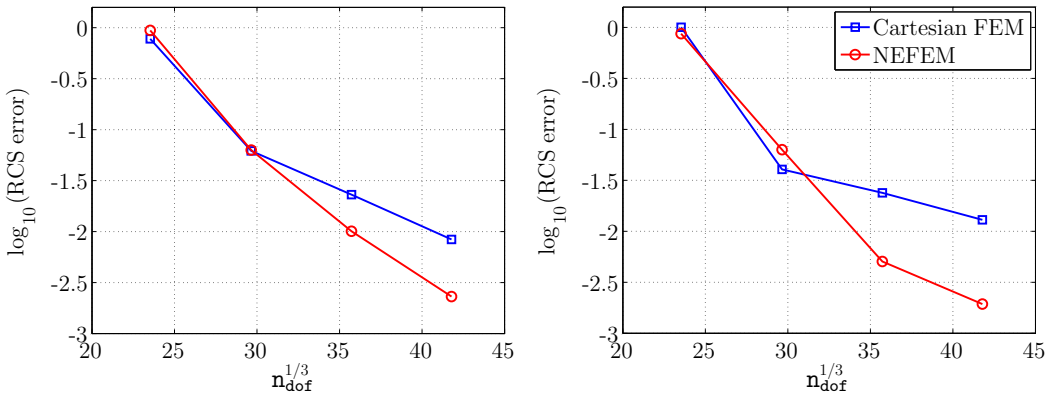
and computed solution is observed with $p = 4$, see Figures 3.19 (b) and 3.20 (b).

Note that cartesian FEs offer a slightly different performance for vertical and horizontal polarizations. In fact, higher errors are observed for the horizontal polarization, whereas for NEFEM, almost identical performance is observed for both polarizations. To compare accuracy, Figure 3.21 represents the RCS error in the $\mathcal{L}^2(-\pi, \pi)$ norm for increasing p, starting with $p = 2$, showing the superiority of NEFEM compared to cartesian FEs. The most critical difference is observed in the horizontal polarization for $p = 5$. NEFEM is almost one order of magnitude more precise than the corre-



(a) Cartesian FEM (b) NEFEM

Figure 3.20: Scattering by a PEC sphere of diameter λ: RCS comparison for increasing p and for the horizontal polarization



(a) Vertical polarization (b) Horizontal polarization

Figure 3.21: Scattering by a PEC sphere of diameter λ: p-convergence comparison of the RCS error

sponding cartesian FEs. It is worth remarking that the RCS error for cartesian FEs is controlled by the geometric error for $p > 3$. In fact, the isoparametric approximation of the sphere with 8 curved elements is considered in Section 2.5, and a similar performance is observed in a second-order elliptic problem. The convergence with NEFEM also deteriorates, for $p > 4$, but it is attributed to errors on the PML. Numerical examples in Appendix E, show that this level of accuracy can not be surpassed if the non-linear PML with thickness λ is considered.

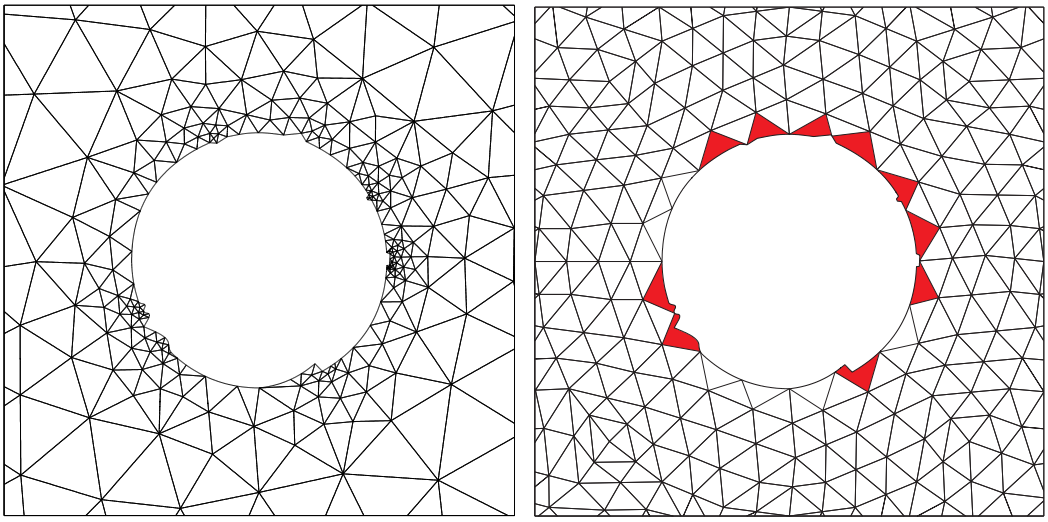
Compared to other techniques NEFEM is also more accurate and efficient. For instance, to achieve an accuracy of 10^{-2} measuring the maximum norm of the scattering width, more than 100 000 degrees of freedom are required using high-order edge elements (Ledger et al., 2003). With NEFEM, a degree of approximation $p = 4$ provides an error of 4.7×10^{-3}, using 45 605 degrees of freedom, that is, NEFEM is two times more accurate by using 50% of the $\mathtt{n_{dof}}$, showing that NEFEM is also competitive in front of other techniques used by the CEM community.

3.3 Small is influential does not imply small elements

Examples in the previous section show the advantages of NEFEM in front of several FE methodologies for the numerical solution of some test cases comparing for the same computational mesh. However, the possibilities of NEFEM still go beyond.

It is well known that, in the context of FEs, the size of the model is sometimes subsidiary of the geometrical complexity and not only on solution itself. In particular, FE simulation of the scattering by complex objects with small geometric details requires drastic h-refinement to capture the geometry. Moreover, for scattering applications, small geometric details are influential in the solution, specially for high frequency problems, and a simplification of the geometry may lead to important discrepancies in the computed scattered field. Nevertheless, as it will be shown next, in the NEFEM context, when small is influential it does not imply small elements.

This section presents three numerical examples that show the possibilities of NEFEM when the scatterer contains small geometric features. As noted earlier, in Section 2.1, it is important to remark that the only restriction for a NEFEM element is that the edges and/or faces on the boundary belong to one NURBS. It is neither necessary to locate nodes at boundary corners or edges (entities with $\mathcal{C}^0$ continuity) nor to refine the mesh near the boundary to capture the geometry. It is exactly represented in NEFEM independently on the spacial discretization. The computational meshes



(a) Refined FEM mesh (b) Coarse NEFEM mesh

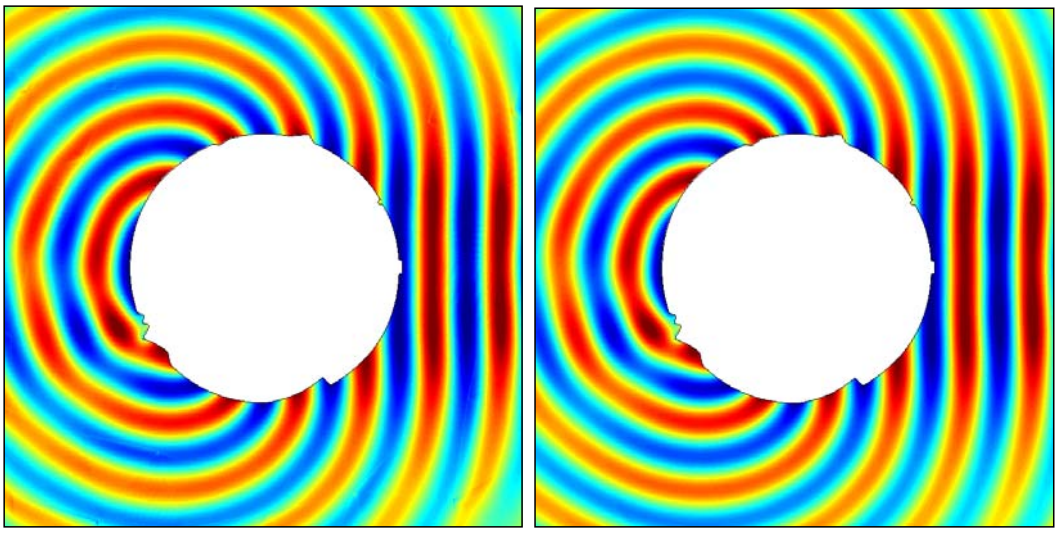
Figure 3.22: Scattering by an irregular circular cylinder of diameter 4λ: detail of a standard FE mesh refined towards the small geometric details, and a coarse NEFEM mesh with elements containing corner geometric singularities

in this section are chosen to emphasize the possibilities of NEFEM.

3.3.1 PEC *irregular* circular cylinder

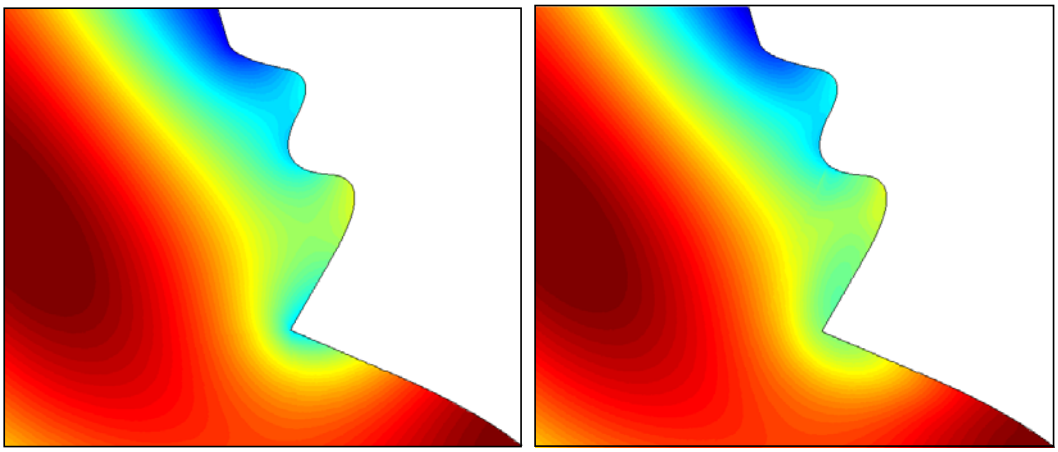
The scattering by a PEC *irregular* circular cylinder of diameter 4λ is considered. Two computational meshes are employed for the analysis, see a detail near the scatterer in Figure 3.22. The first mesh is a standard FE mesh in which h-refinement is performed in order to provide an accurate description of the small geometric features, see Figure 3.22 (a). The resulting mesh has 130 curved elements. The second mesh, represented in Figure 3.22 (b), is a coarse *NEFEM mesh* with only 16 curved elements, some of them (represented in red) containing small geometric details and corner singularities inside an edge.

Figure 3.23 shows the transverse field H_3 computed in the refined mesh with a degree of interpolation $p = 5$, and in the coarse mesh with a degree of interpolation $p = 12$. The scattered fields are indistinguishable, even near the most critical zone depicted in Figure 3.24. Nevertheless, a slight difference near the corner singular-



(a) Refined FEM mesh (b) Coarse NEFEM mesh

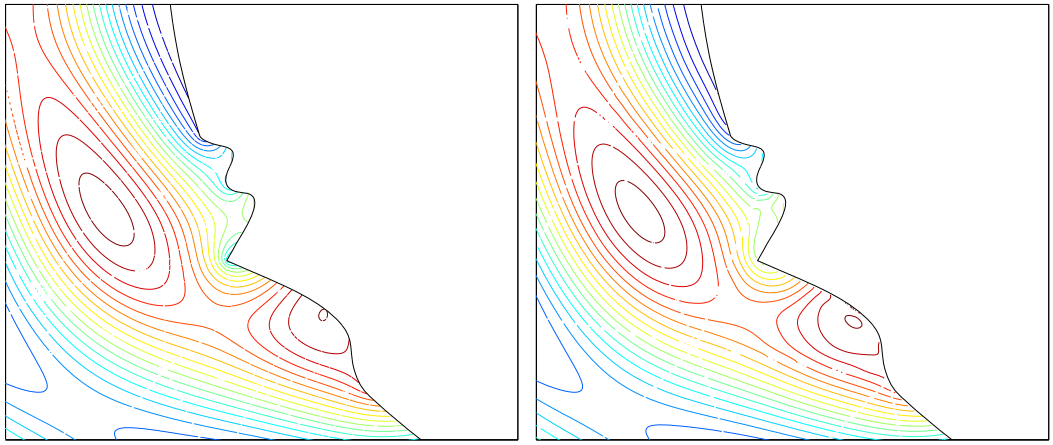
Figure 3.23: Scattering by an irregular circular cylinder of diameter 4λ: H_3 field computed in the discretizations shown in Figure 3.22



(a) Refined FEM mesh (b) Coarse NEFEM mesh

Figure 3.24: Scattering by an irregular circular cylinder of diameter 4λ: detail of the H_3 fields shown in Figure 3.23

ity can be observed representing the isolines of the scattered field, see Figure 3.25. Obviously, the discrepancy is originated by the limitations of the standard FE nodal interpolation for the approximation of a singular solution, see Szabó and Babuška (1991) and following examples in this section. Despite of this know limitation, it is



(a) Refined FEM mesh (b) Coarse NEFEM mesh

Figure 3.25: Scattering by an irregular circular cylinder of diameter 4λ: isolines of the H_3 fields shown in Figure 3.24

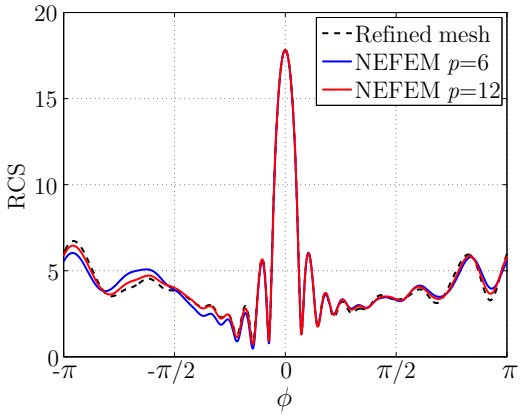
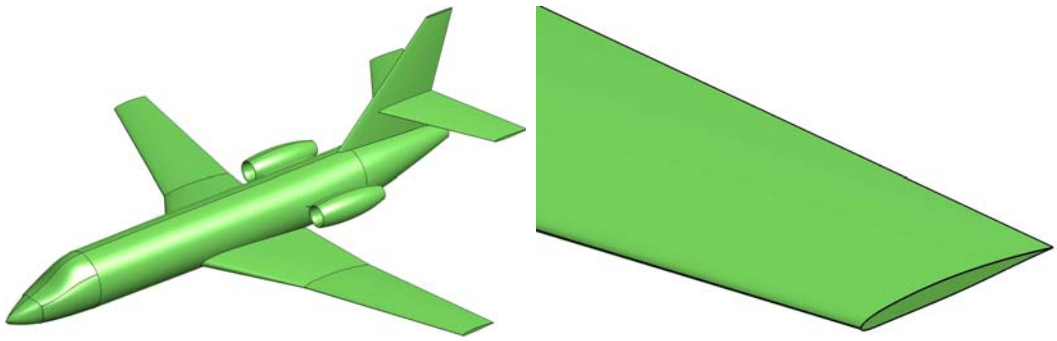


Figure 3.26: Scattering by an irregular circular cylinder of diameter 4λ: RCS comparison

important to remark that the quantity of interest, the RCS, shows very good agreement when it is compared with the RCS computed with the refined mesh, see Figure 3.26. In fact, two NEFEM computations are performed in the coarse mesh of Figure 3.22 (b), with $p = 6$ and $p = 12$, illustrating the convergence as the degree of approximation is increased. For $p = 12$ the relative RCS error in the $\mathcal{L}^2(-\pi, \pi)$ norm is 4.1×10^{-2}.

Finally, it is worth mentioning that the drastic difference between minimum mesh



(a) (b)

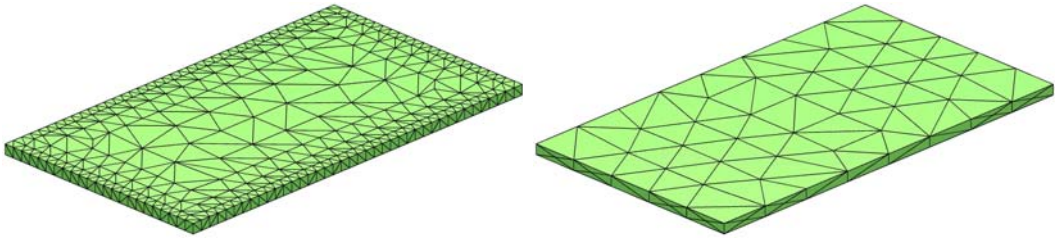
Figure 3.27: (a) NURBS surfaces of a complete aircraft and (b) detailed view near a wing

sizes, in the discretizations shown in Figure 3.22, induces important differences in the time-step size when explicit time integrators are used and, therefore, adds another advantage of NEFEM. In the refined mesh the minimum distance between two mesh nodes is 1.4×10^{-5} with $p = 5$, whereas in the NEFEM mesh the minimum distance is 1.2×10^{-3} with $p = 12$. The computation with the refined mesh requires 527 459 time steps, whereas the computation in the coarse NEFEM mesh requires 6 620 time steps.

3.3.2 PEC *thin* plate

The aim of the following example is to show the possibilities of *NEFEM elements*, containing edge singularities, in 3D domains. In large scale 3D computations, very small geometric details may lead to unaffordable computational times with explicit time-marching algorithms, due to the excessive h-refinement needed to accurately capture the geometry. For instance, consider the complete aircraft represented in Figure 3.27 (a). A detailed view near a wing is represented in Figure 3.27 (b), revealing the dramatically small thickness of the wing compared to the total length of the aircraft. Thus, standards FEs require excessive refinement to accurately represent the geometry.

To show the capabilities of NEFEM in this scenario, the scattering by a PEC *thin*



(a) Refined FEM mesh (b) FEM mesh

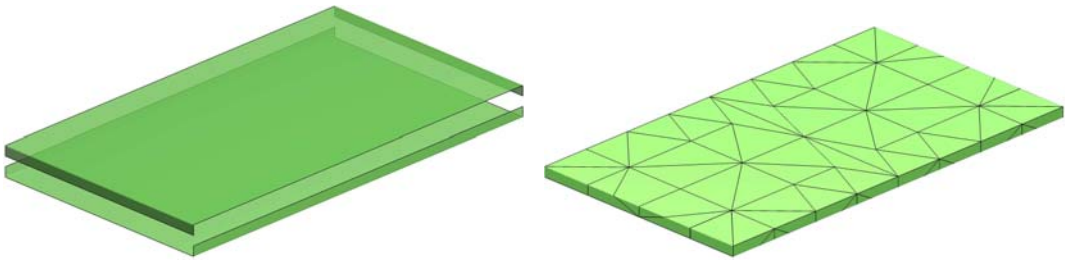
Figure 3.28: Scattering by a PEC thin plate: standard FE meshes

plate of dimensions $\lambda \times 4\lambda/7 \times \lambda/22$ is considered. Note that the plate is only one wavelength long, but for a wing of the same thickness, this frequency corresponds to a complete aircraft of characteristic length 20λ. The small thickness of the plate, with respect to the wave length λ, implies that h-refinement in standard FE meshes is controlled by the thickness of the plate, not by the desired number of nodes per wavelength.

Two standard FEM computational meshes are considered to compare the accuracy of NEFEM computations. Figure 3.28 (a) shows a standard FE mesh with refinement towards the singularities of the plate. The second mesh, in Figure 3.28 (b), is a FEM mesh with a desired mesh size of about $\lambda/8$. As usual, a standard mesh generator needs to perform extra h-refinement to offer an accurate description of the geometrical model. Therefore, the minimum mesh size in a standard FE mesh is, at least, $\lambda/22$.

Nevertheless, the mesh size for NEFEM is not controlled by small geometric features, and the desired mesh size is maintained, even in the presence of singularities in the boundary of the domain. The plate is exactly represented by two B-spline surfaces with $\mathcal{C}^0$ continuity at the edges of the plate, as illustrated in Figure 3.29 (a). A NEFEM coarse mesh is represented in Figure 3.29 (b). Note that, to obtain the desired mesh size, some elements contain an edge singularity inside one NURBS face, see a detailed view of a NEFEM element in Figure 3.30.

Figure 3.31 compares the RCS distribution for vertical and horizontal polarizations. An excellent agreement is observed between the three computations, showing the potential of NEFEM coarse meshes with elements containing singularities. Again,



(a) NURBS surfaces (b) NEFEM mesh

Figure 3.29: Scattering by a PEC thin plate: NURBS surfaces (separated for visualization) and NEFEM coarse mesh with elements containing edge singularities

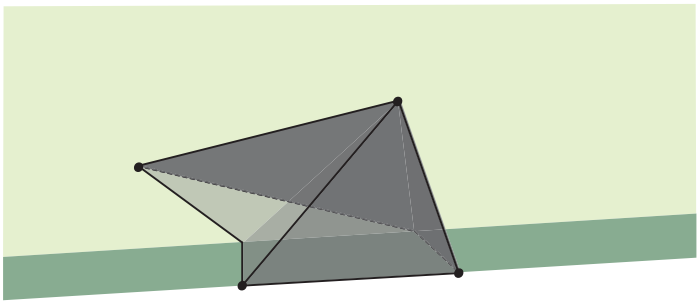
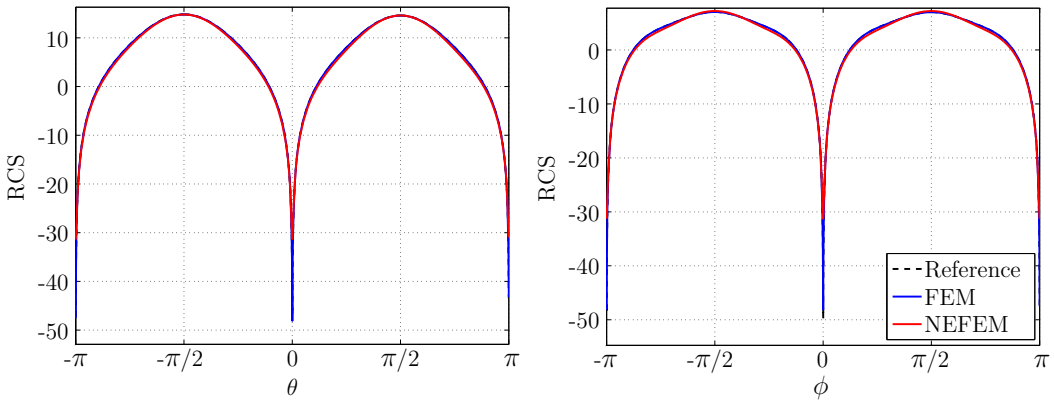


Figure 3.30: Detailed view of a NEFEM element containing an edge singularity in its boundary face

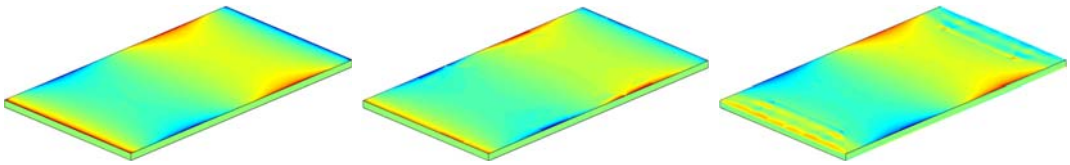
the maximum disagreement is obtained at singularities of the RCS due to its logarithmic scale. The error of the scattering width and in the $\mathcal{L}^2(-\pi,\pi)$ norm for NEFEM is 3.2×10^{-2} and 4.7×10^{-2} for the vertical and horizontal polarizations respectively.

Figure 3.32 compares the scattered E_3 field computed with FEM in the standard meshes shown in Figure 3.28, and with NEFEM in the discretization shown in Figure 3.29. In each case a degree of approximation $p = 4$ is used. Good agreement is obtained in the scattered field distributions, but some differences are observed near the singularities of the plate, for both FEM and NEFEM solutions in the coarse meshes shown in Figures 3.28 (b) and 3.29 (a) respectively. As expected, refinement towards the singularity allows to isolate the effect of the singularity and to accurately represent the field values at the edges of the plate. With the FEM mesh of Figure 3.28 (b)



(a) Vertical polarization (b) Horizontal polarization

Figure 3.31: Scattering by a PEC thin plate: comparison of the RCS computed in the discretizations shown in Figures 3.28 (a), 3.28 (b) and 3.29 (a)



(a) Refined FEM mesh (b) FEM mesh (c) NEFEM mesh

Figure 3.32: Scattering by a PEC thin plate: scattered E_3 field over the plate

less quality of the scattered field is observed near the edges of the plate, although the minimum mesh size is approximately the same than in the refined mesh. With NEFEM meshes the nodal interpolation is not able to accurately describe the singularity inside an element face, and the Runge's phenomena is observed in the approximation of the field represented in Figure 3.32 (c). This discrepancy shows, again, a limitation of the standard FE nodal approximation near singularities. Nevertheless, it is worth recalling the important save in computational memory and cost with NEFEM meshes, and the excellent agrement obtained in the RCS.

3.3.3 L-shaped model problem

As shown in the previous examples, h-refinement is not needed in NEFEM to accurate capture the geometry, even in the presence of corner or edge singularities. Nevertheless, it is well known that, in such cases, the electric field may be unbounded (Costabel and Dauge, 1997) and the standard nodal FE approximation may lead to a non-optimal p-convergence, see also Szabó and Babuška (1991) and previous examples in this section.

The performance of NEFEM elements with a corner inside one edge is studied using the scattering by a L-shaped domain, see Ledger et al. (2003) and references therein. The time-dependent analytical solution is defined as

$$\boldsymbol{E}(t) = Re(\widehat{\boldsymbol{E}} e^{jwt})$$

where

$$\widehat{\boldsymbol{E}} = \Big(\frac{\partial f}{\partial y}, -\frac{\partial f}{\partial x}\Big), \qquad f = J_\alpha(\omega r)\cos(\alpha\phi).$$

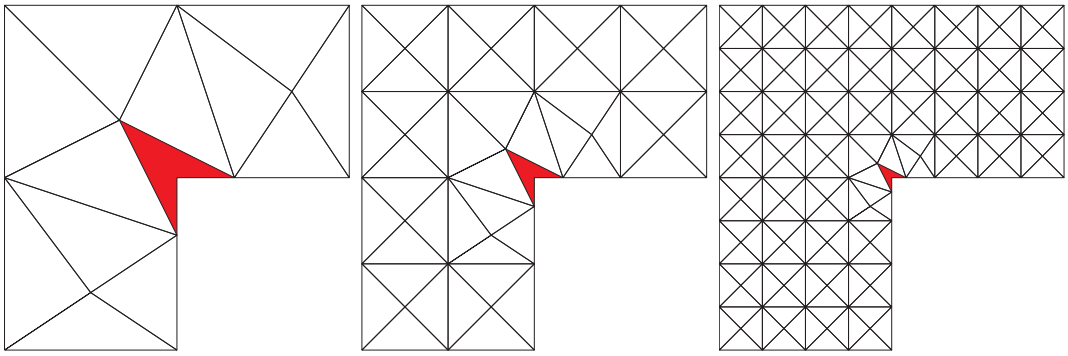
Here, J_α is the Bessel function of the first kind and order α, $\omega = 2\pi/\lambda$ is the angular frequency, (r, ϕ) are the polar coordinates, and the tilde denotes the complex amplitudes of the fields in the frequency domain. As usual, the magnetic field is defined in terms of the electric field by

$$\boldsymbol{H}(t) = Re(\widehat{\boldsymbol{H}} e^{jwt}),$$

with

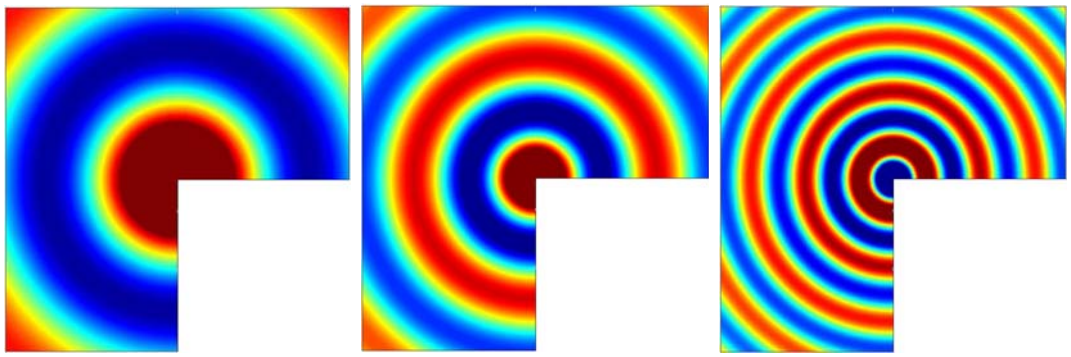
$$\widehat{\boldsymbol{H}} = -\frac{1}{jw\mu}\nabla \times \widehat{\boldsymbol{E}}.$$

Note that the order α of the Bessel function allows to define both smooth and singular analytical solutions. For instance, with $\alpha = 0$ the fields are smooth, and choosing a value $\alpha = 2/3$ the solution exhibits a singularity at the origin. Dirichlet boundary conditions corresponding to the analytical solution are applied in the whole boundary.



(a) Mesh 1 (b) Mesh 2 (c) Mesh 3

Figure 3.33: L-shaped model problem: NEFEM meshes with an element containing a corner singularity

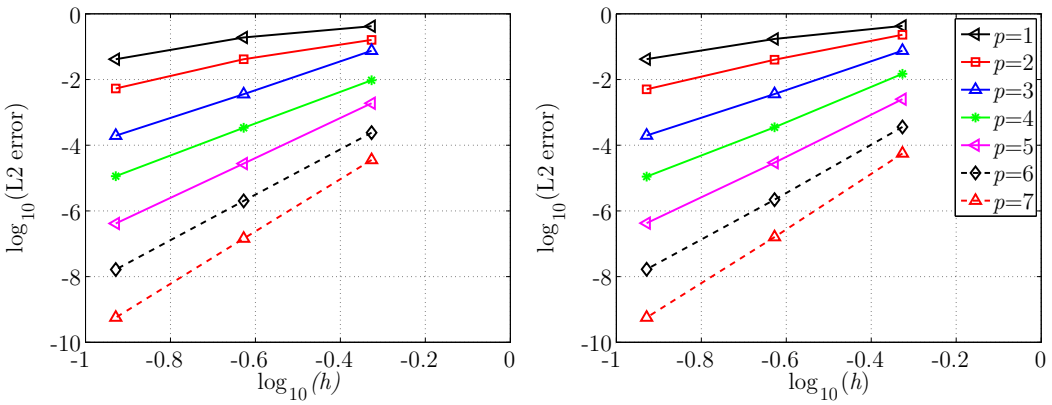


(a) Mesh 1, $\omega = 5$ (b) Mesh 2, $\omega = 10$ (c) Mesh 3, $\omega = 20$

Figure 3.34: L-shaped model problem: H_3 field for a NEFEM solution with $p = 5$

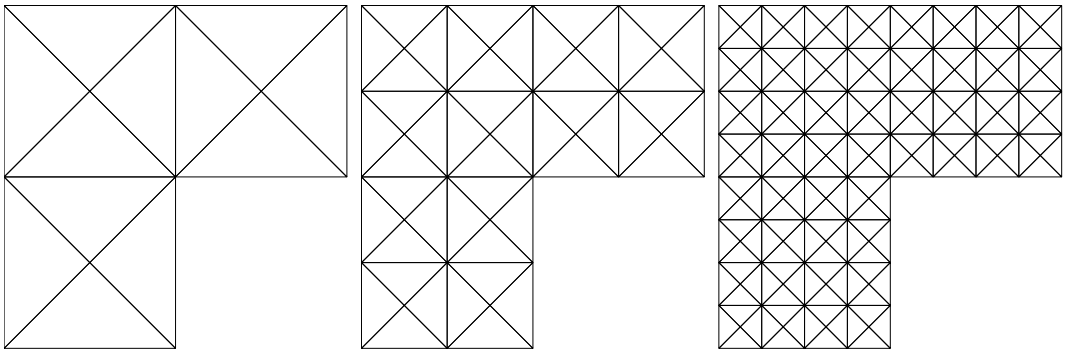
Figure 3.33 shows three computational meshes for NEFEM. Each mesh contains an element, represented in red, with a corner inside an edge. In the computations, only the element in red is considered a *curved element*, and its *curved boundary* is defined as a piecewise polynomial B-spline parametrization with $\mathcal{C}^0$ continuity at the origin. The transverse field, H_3, for a NEFEM solution with $p = 5$ is shown in Figure 3.34 at different frequencies. Each plot corresponds to an increasing value of the angular frequency, namely $\omega = 5$, $\omega = 10$ and $\omega = 20$, and are computed with the meshes shown in Figure 3.33.

Optimal h-convergence rates of NEFEM are obtained for $\alpha = 0$, i.e. for a smooth analytical solution. Figure 3.35 shows the $\mathcal{L}^2(\Omega)$ norm of the error of the H_3 field



(a) FEM meshes (b) NEFEM meshes

Figure 3.35: L-shaped model problem: h-convergence for standard FE meshes and for NEFEM meshes with a corner inside an edge

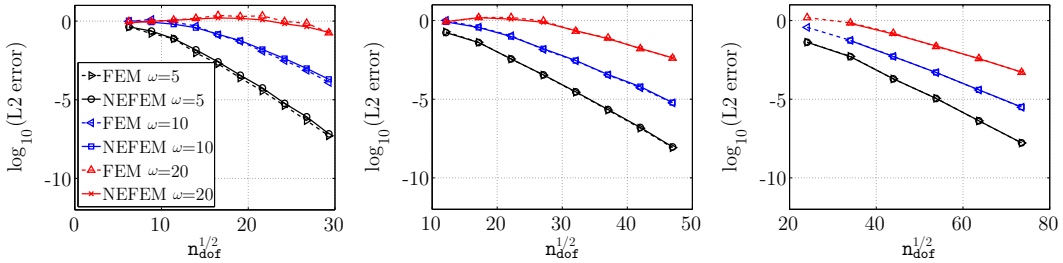


(a) Mesh 1 (b) Mesh 2 (c) Mesh 3

Figure 3.36: L-shaped model problem: standard FEM meshes

under h-refinement, for $\alpha = 0$, $\omega = 5$, with the meshes of Figure 3.33 and using a polynomial approximation of degree up to $p = 7$. It is important to note that, not only the optimal rate of h-convergence is obtained using NEFEM meshes with a corner inside an edge, but also the error with NEFEM is almost identical to the error with standard FEM meshes, represented in Figure 3.36.

Next, a p-convergence comparison is presented between standard FEM meshes and NEFEM meshes. Figure 3.37 shows the evolution of the $\mathcal{L}^2(\Omega)$ error of the H_3 field, as p is uniformly increased starting with $p = 1$. In both cases, optimal (exponential)



(a) Mesh 1 (b) Mesh 2 (c) Mesh 3

Figure 3.37: L-shaped model problem: p-convergence comparison on the meshes shown in Figures 3.36 and 3.33 (a)

convergence rate is observed, as expected for a problem with a smooth solution.

Numerical experiments with corner singularities and smooth analytical solutions show an optimal performance of NEFEM meshes compared to standard FEM meshes for the same $\mathtt{n_{dof}}$. Thus, when the singularity on the domain does not imply a singularity in the solution, the use of NEFEM is a powerful strategy to use coarse meshes and high-order approximations to properly capture the solution, avoiding mesh refinement near small geometric details.

When the solution is singular, for instance if $\alpha = 2/3$ in this example, the optimal rate of convergence is not achieved with uniform h or p-refinement even with standard FEM meshes (Szabó and Babuška, 1991). To illustrate the behavior in the presence of singularities, a p-refinement comparison is performed with the FEM mesh shown in Figure 3.36 (a). The H_3 error for increasing p is depicted in Figure 3.38 for a problem with a smooth solution ($\alpha = 0$) and for a problem with a singular solution ($\alpha = 2/3$). The optimal (exponential) rate of convergence is not achieved for the problem with singular solution. In fact, only an algebraic rate of convergence is observed, see Szabó and Babuška (1991) for more details. With NEFEM meshes the situation is slightly different, because the singularity does not coincide with a node of the mesh, but, in any case, the exponential convergence is not achieved.

The potential benefits of NEFEM methodology requires further study in this direction, see comments in Chapter 4.

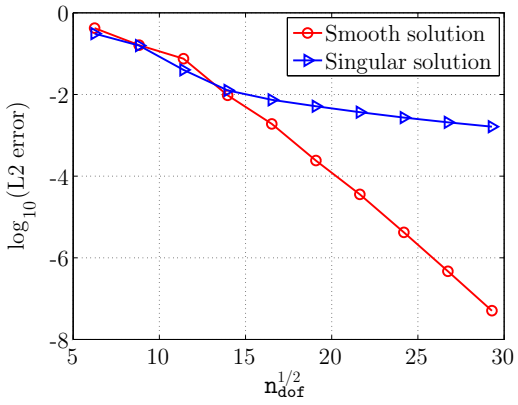


Figure 3.38: L-shaped model problem: p-convergence comparison for smooth and singular solutions on the meshes shown in Figure 3.36 (a)

3.4 Challenging applications

The challenges of solving Maxwell's equations are not typically found in the equations, but in the geometrical complexity of the scatterer and/or in the wavelength of the incident field (with respect to the characteristic length of the scatterer).

This section presents some examples in which the challenge is given by a complex geometric scatterer and/or a relative high frequency, considering 2D and 3D obstacles of characteristic length up to 21λ.

3.4.1 PEC Sphere

The following example considers the scattering of an incident wave travelling in the z^+ direction by a PEC sphere of diameter 20λ. The mesh used in the computations has 124 135 elements with planar faces and 17 856 curved elements (11 176 elements with a face on the NURBS boundary and 6 680 elements with an edge on the NURBS boundary). The surface mesh of the sphere is represented in Figure 3.39. The scattered electric field over the surface of the sphere is represented in Figure 3.40 for a NEFEM solution with $p = 4$. A comparison of the computed RCS with the analytical solution is depicted in Figure 3.41, showing good agreement with the analytical solution. In fact, the RCS distributions overlap for a viewing angle in $[-\pi/4, \pi/4]$,

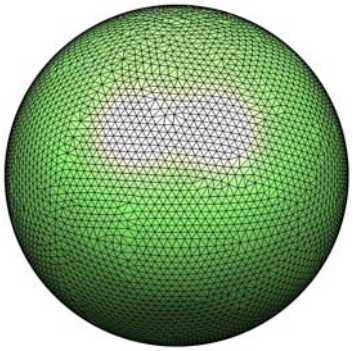
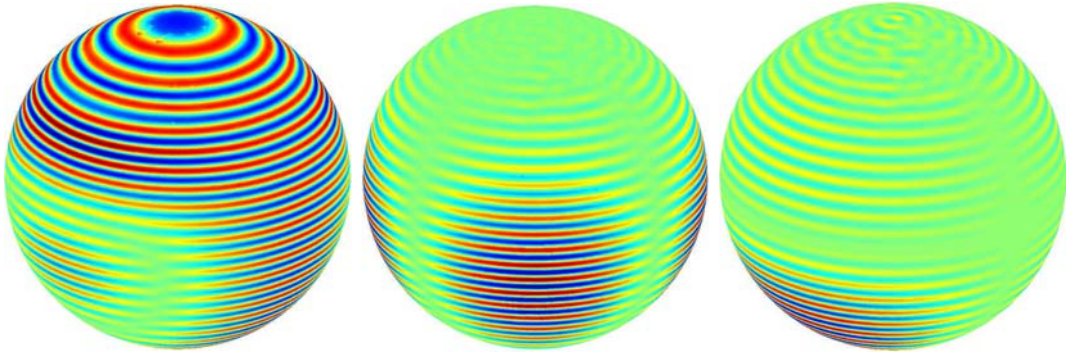
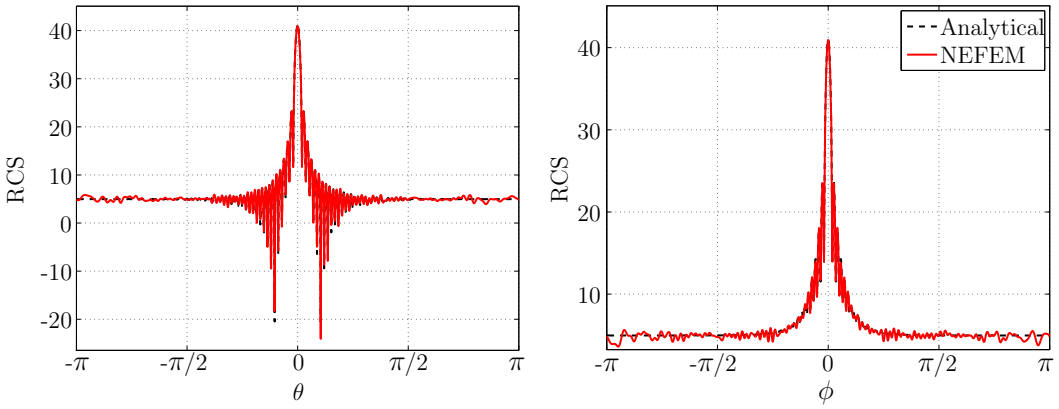


Figure 3.39: Scattering by a PEC sphere of diameter 20λ: computational mesh



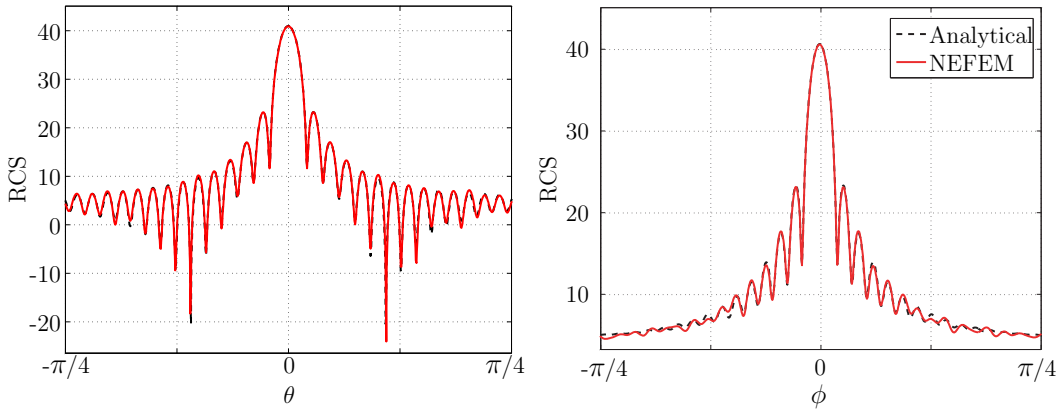
(a) E_1 (b) E_2 (c) E_3

Figure 3.40: Scattering by a PEC sphere of diameter 20λ: scattered electric field for a NEFEM solution with $p = 4$



(a) Vertical polarization (b) Horizontal polarization

Figure 3.41: Scattering by a PEC sphere of diameter 20λ: RCS for a NEFEM solution with $p = 4$

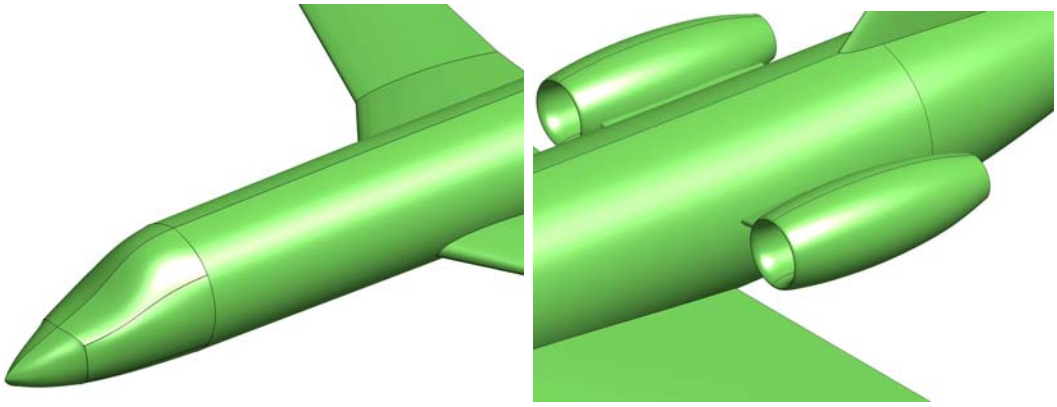


(a) Vertical polarization (b) Horizontal polarization

Figure 3.42: Scattering by a PEC sphere of diameter 20λ: RCS for a NEFEM solution with $p = 4$ in the range $[-\pi/4, \pi/4]$

see Figure 3.42, and some differences are observed in the other viewing angles. Note that the spatial discretization is relatively coarse for this frequency and a reasonable accuracy for engineering purposes is obtained. The relative RCS in the $\mathcal{L}^2(-\pi, \pi)$ norm is 4.7×10^{-2} for the vertical polarization and 6.3×10^{-2} for the horizontal polarization. Again, the error in the RCS is higher for the horizontal polarization due to the singularity induced by the logarithmic scale. If the error is measured in the scattering width, an error of 1.2×10^{-2} is obtained for both polarizations. If lower errors are required, further p-refinement can be performed on the same mesh.

NEFEM is also a competent approach to 3D challenging simulations compared to other techniques. For instance, the method proposed by Huttunen et al. (2007) is applied to compute the scattering by a conducting sphere in the frequency domain. To achieve a relative error of about 10^{-2} for a sphere of diameter 15λ, almost 4 million of degrees of freedom are needed. In this section, same accuracy is obtained with NEFEM for a higher frequency problem (sphere of diameter 20λ) using 4 344 725 degrees of freedom. It is also worth mentioning that a improved PML is applied by Huttunen et al. (2007) in the frequency domain, that allows a 0.2λ thick PML, whereas the NEFEM computation uses a λ thick PML, demonstrating, once more, the benefits of NEFEM.



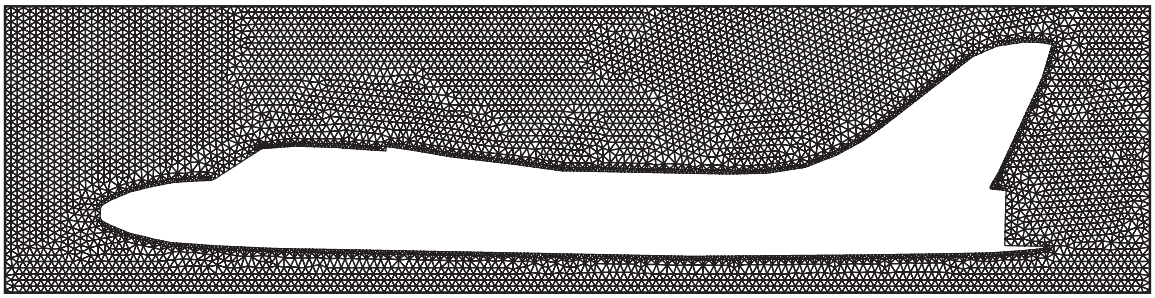
(a) (b)

Figure 3.43: NURBS surfaces of a complete aircraft: detailed view of (a) the front part of the aircraft and leading edge of the wings, and (b) the engine air intakes

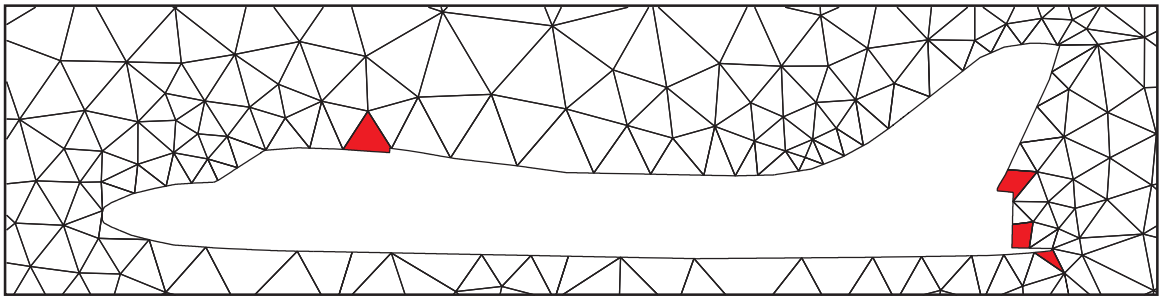
It is also worth remarking that even if the surface mesh of the obstacle is refined for high frequency applications, an accurate geometric model is still important. On one hand, as noted by Huttunen et al. (2007), as the frequency is increased, geometric errors are more influential in the scattered field. For instance, with isoparametric FEs, the geometric singularities introduced at boundary edges may produce non physical diffraction. On the other hand, as noted in previous section, small geometric details may lead to extremely refined meshes. Even for higher frequencies, the mesh of a complex geometric obstacle may lead to further refinement to capture small geometric features. For instance, Figure 3.43 shows a detailed view of two zones of the aircraft represented in Figure 3.27, showing abrupt variations of the outward normal in the front part of the aircraft, in the leading edge of the wings and in the engine air intakes.

3.4.2 PEC aircraft profile

The following example considers the scattering by a PEC aircraft profile of length 10λ. A comparison between low-order elements and high-order *NEFEM elements* is performed, with the computational meshes shown in Figure 3.44. The fine mesh is a standard FEM mesh, with 33 338 elements (32 737 straight-sided elements and 601



(a) FEM mesh



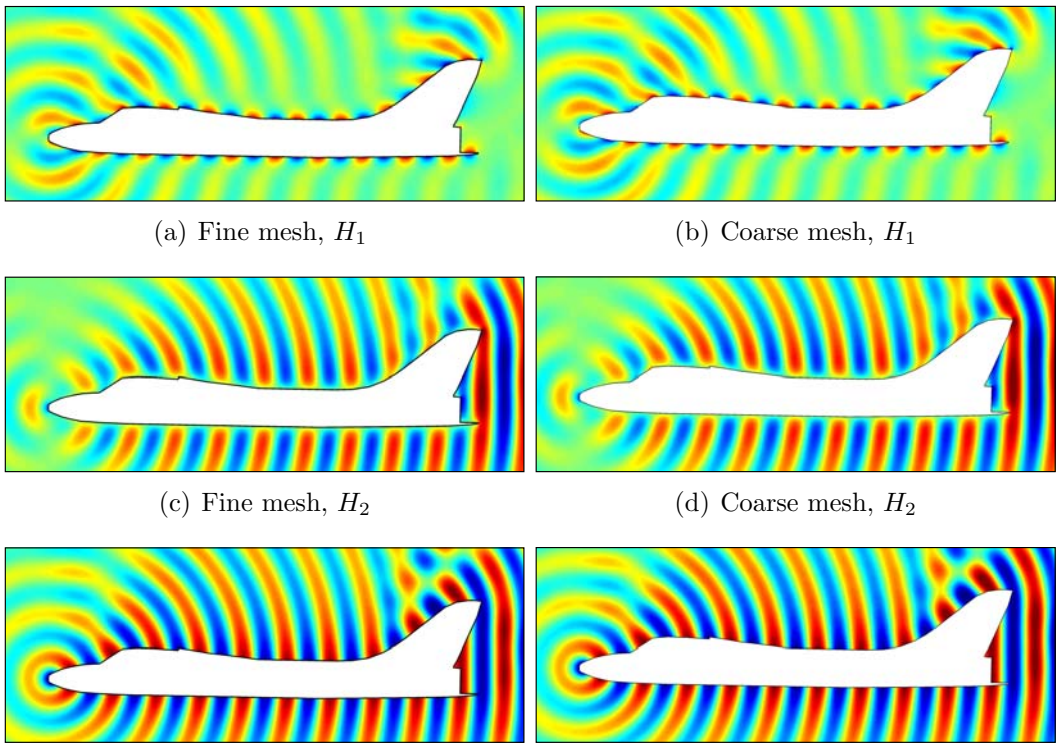
(b) NEFEM mesh

Figure 3.44: Scattering by an aircraft profile of length 10λ: (a) refined mesh for a low-order computation, and (b) coarse mesh with elements containing corner for a NEFEM computation with high-order approximations

curved elements) and a total of 200 028 nodes for a degree of interpolation $p = 2$. The coarse mesh has 725 elements (674 straight-sided elements and 51 curved elements), with a total of 32 625 nodes for a degree of interpolation $p = 8$.

The geometry of the 2D section of this aircraft has several critical zones, in particular, a small irregularity on the upper part and the rear part. Note that the spatial discretization for NEFEM is independent on the complex geometric details and it is designed to accurately capture the solution and to emphasize the performance of NEFEM elements in red. Thus, some elements, colored in Figure 3.44 (b), contain corner singularities inside their edges.

Figure 3.45 represents the scattered fields for quadratic isoparametric FEs in the mesh of Figure 3.44 (a), and for NEFEM with $p = 8$ in the coarse mesh of Figure 3.44 (b), showing an excellent agreement between both solutions. To compare the solution near the most critical zones, two detailed views of the scattered fields are shown in



(a) Fine mesh, H_1 (b) Coarse mesh, H_1

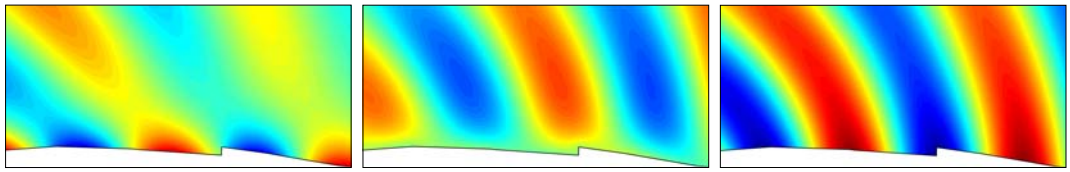
(c) Fine mesh, H_2 (d) Coarse mesh, H_2

(e) Fine mesh, E_3 (f) Coarse mesh, E_3

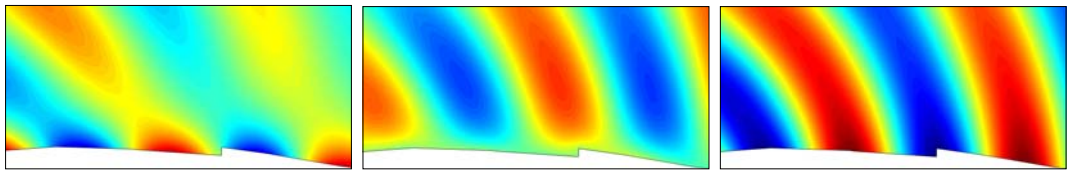
Figure 3.45: Scattering by an aircraft profile of length 10λ: comparison of the scattered fields obtained in the fine and coarse meshes

Figures 3.46 and 3.47. An excellent agreement is observed for all the components of the scattered field. On the upper part of the aircraft the results are indistinguishable, even in the vicinity of the singular points. On the rear part of the aircraft, a perfect agreement is also observed, but small differences can be observed in the vicinity of one of the corner singularities. In fact, it is worth to note that main differences are observed in a NEFEM element containing two corner singularities, that has been introduced to emphasize the possibilities of NEFEM, see the mesh in Figure 3.44 (b), not in the vicinity of the NEFEM element containing the smallest geometric detail.

Figure 3.48 compares the RCS obtained in each case, showing a perfect match, and therefore, demonstrating the capabilities of NEFEM to compute accurate solutions with elements containing corner singularities in more complex situations. The difference between both RCS patterns measured in the $\mathcal{L}^2(-\pi,\pi)$ norm is 1.5×10^{-2}.

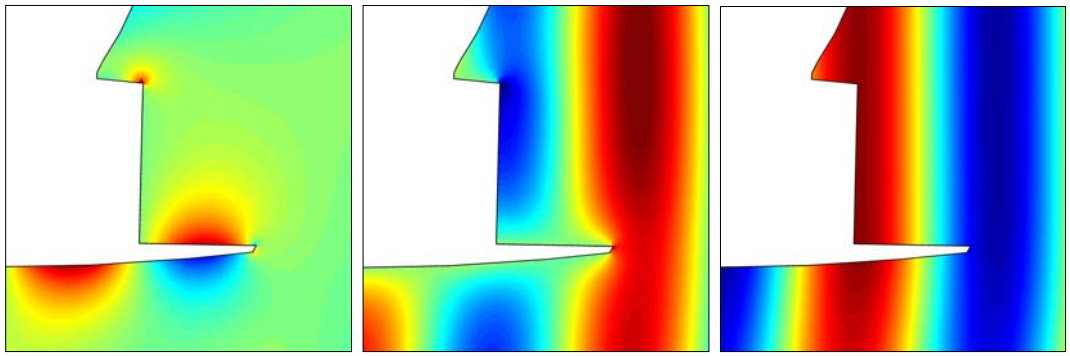


(a) Fine mesh, H_1 (b) Fine mesh, H_2 (c) Fine mesh, E_3

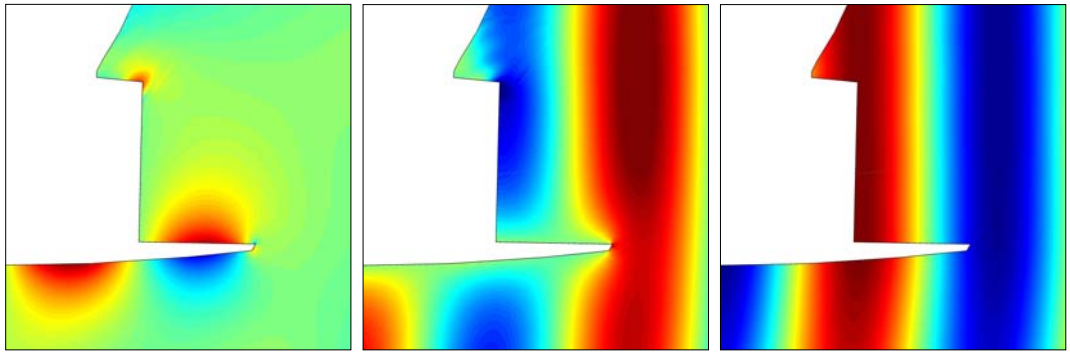


(d) Coarse mesh, H_1 (e) Coarse mesh, H_2 (f) Coarse mesh, E_3

Figure 3.46: Scattering by an aircraft profile of length 10λ: comparison of the scattered fields obtained in the fine and coarse meshes at the upper part of the aircraft



(a) Fine mesh, H_1 (b) Fine mesh, H_2 (c) Fine mesh, E_3



(d) Coarse mesh, H_1 (e) Coarse mesh, H_2 (f) Coarse mesh, E_3

Figure 3.47: Scattering by an aircraft profile of length 10λ: comparison of the scattered fields obtained in the fine and coarse meshes at the rear part of the aircraft

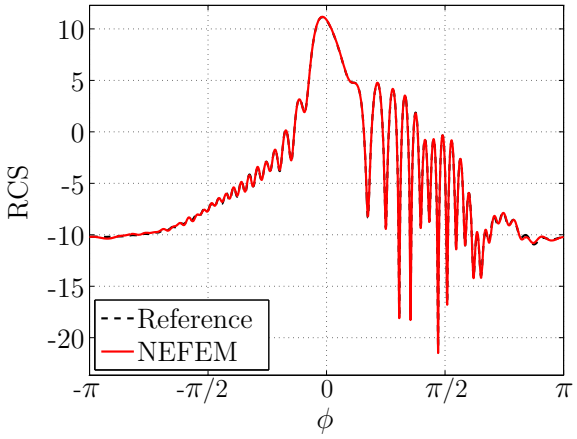


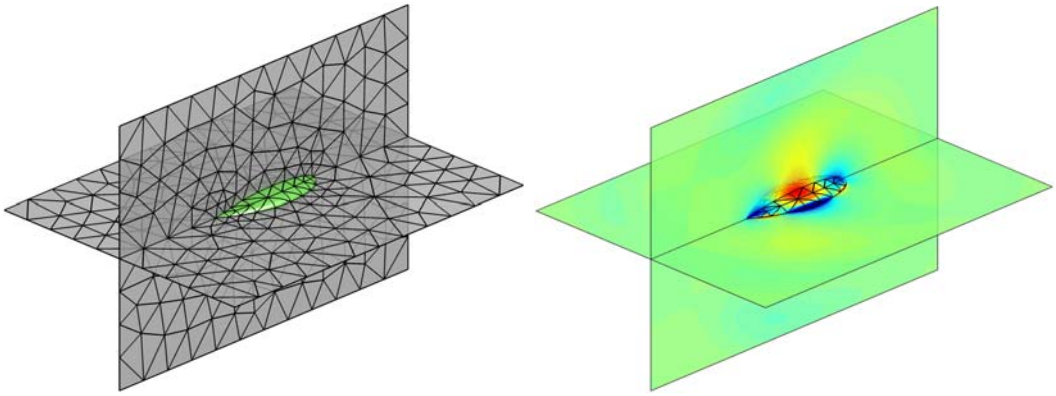
Figure 3.48: Scattering by an aircraft profile of length 10λ: comparison of the RCS in the fine and coarse meshes

It is worth remarking that using classical isoparametric FE it is not possible to compute accurate solutions for these problems with the computational meshes used by NEFEM, see Figure 3.44 (b). To properly capture the geometry of the domain with isoparametric FE it is necessary to discretize accounting for corners nodes (boundary points with only $\mathcal{C}^0$ continuity).

3.4.3 PEC NASA almond

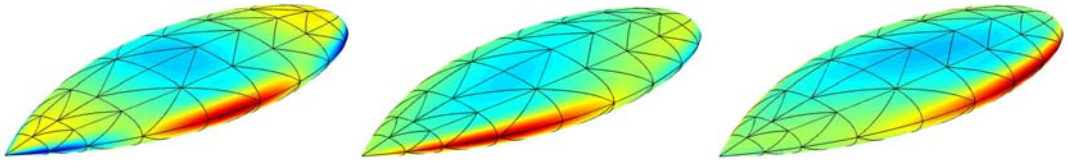
The following example considers a popular benchmark for 3D RCS computations, the scattering by a PEC NASA almond, see Dominek and Shamanski (1990) or Woo et al. (1993). One of the challenges of this example is the singularity exhibited by the solution on the tip of the almond. Moreover, the high variation on the surface curvature introduces extra complexity.

First, the monostatic RCS computation of an almond of characteristic length λ is considered. The mesh employed for the computation has 10 805 elements with planar faces and 336 curved elements (120 with a face on the NURBS boundary, and 216 with an edge on the NURBS boundary). Figure 3.49 (a) shows two cuts of the volume mesh, corresponding to the free-space, and the surface mesh of the almond.



(a) (b)

Figure 3.49: Scattering by a PEC NASA almond of characteristic length λ: (a) two cuts of a coarse mesh, and (b) H_2 field for a NEFEM solution with $p = 4$



(a) $\phi^i = 0$ (b) $\phi^i = \pi/3$ (c) $\phi^i = 2\pi/3$

Figure 3.50: Scattering by a PEC NASA almond of characteristic length λ: E_3 field for different angles of incidence for a NEFEM solution with $p = 4$

The scattered H_2 field for a NEFEM solution with $p = 4$ is represented in Figure 3.49 (b), for a wave incident onto the tip of the almond.

The monostatic RCS evaluation is performed by computing $N^i = 18$ solutions corresponding to a series of incident angles $\phi^i_j = j\pi/N^i$, with $j = 0, \ldots, N^i$. Figure 3.50 shows the scattered E_3 field on the surface of the almond for three different angles of incidence.

The monostatic RCS for the vertical polarization is represented in Figure 3.51, and compared with a reference solution, showing excellent agreement. The difference between both solutions is 1.5×10^{-2} in the $\mathcal{L}^2(-\pi, \pi)$ norm. The reference data corresponds to the published results by Ledger et al. (2007), which are obtained using high-order edge elements with non-uniform degree of approximation on a tetrahedral mesh with 4 723 elements. In the NEFEM computation, the markers correspond to

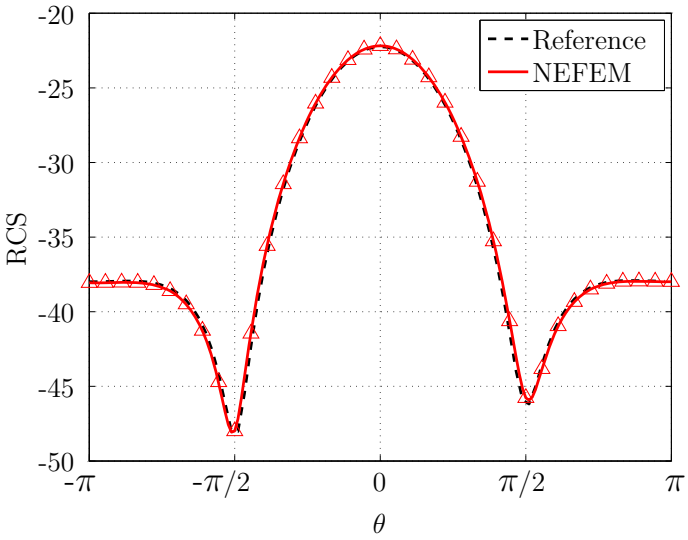
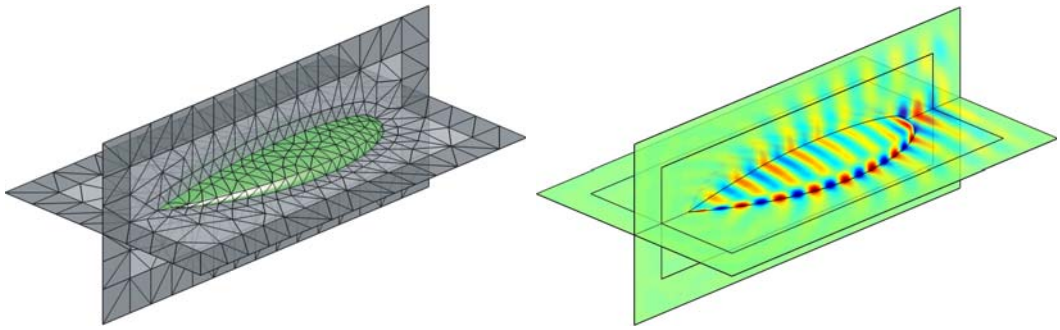


Figure 3.51: Scattering by a PEC NASA almond of characteristic length λ: monostatic RCS for a NEFEM solution with $p = 4$, compared with high-order edge elements of Ledger et al. (2007)

the 18 computations, and the continuous line corresponds to a postprocess of the monostatic data, as described by Schuh et al. (1994).

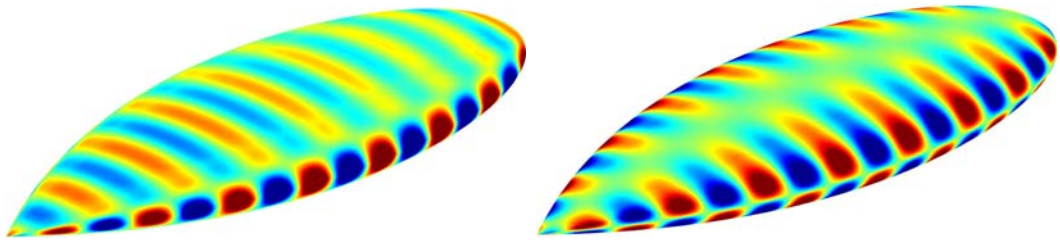
Next example considers the scattering of a plane electromagnetic wave by a PEC NASA almond of characteristic length 8λ. The mesh employed for the computation has 9 348 elements with planar faces and 1 200 curved elements (636 with a face on the NURBS boundary, and 564 with an edge on the NURBS boundary). Figure 3.52 (a) shows two cuts of the volume mesh, showing a λ thick PML, and the surface mesh of the almond. The scattered E_3 field for a NEFEM solution with $p = 5$ is represented in Figure 3.52 (b), for a wave incident onto the tip of the almond. A detailed view of two components of the magnetic field, namely H_2 and H_3, are represented over the surface of the almond in Figure 3.53.

Figure 3.54 shows the bistatic RCS for the vertical and horizontal polarizations. Two RCS patterns are displayed, for a NEFEM solution with $p = 4$ and $p = 5$ respectively. The results compare well with published results by Hachemi et al. (2004). In this work, linear FE in a continuous Galerkin framework are considered. The tetrahedral mesh used by Hachemi et al. (2004) has 20 188 surface triangles and 1 121 431



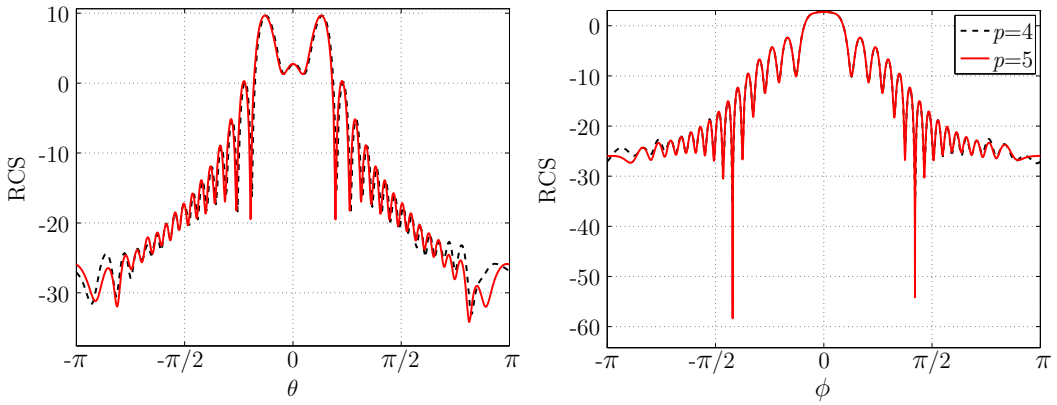
(a) (b)

Figure 3.52: Scattering by a PEC NASA almond of characteristic length 8λ: (a) two cuts of a tetrahedral mesh, and (b) E_3 field for a NEFEM solution with $p = 5$



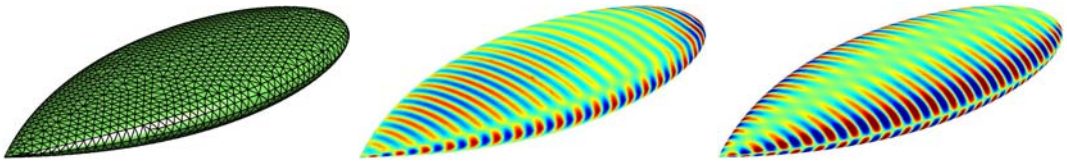
(a) H_2 (b) H_3

Figure 3.53: Scattering by a PEC NASA almond of characteristic length 8λ: two components of the magnetic field for a NEFEM solution with $p = 5$



(a) Vertical polarization (b) Horizontal polarization

Figure 3.54: Scattering by a PEC NASA almond of characteristic length 8λ: RCS for a NEFEM solution with $p = 4$ and $p = 5$



(a) Surface mesh (b) E_3 (c) H_3

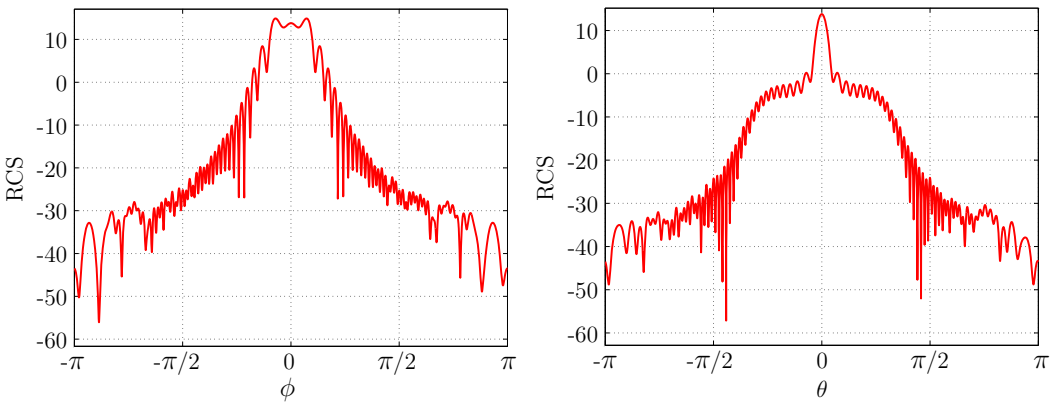
Figure 3.55: Scattering by a PEC NASA almond of characteristic length 21λ: surface mesh of the almond and two components of the scattered field for a NEFEM solution with $p = 3$

mesh nodes, whereas the NEFEM computation with $p = 5$ uses 590 688 nodes. Thus, this example shows the competitiveness of NEFEM in front of other formulations for more challenging applications. Even if a DG formulation is considered, i.e. duplicating nodes at inter-element faces, the computation requires less degrees of freedom to obtain similar accuracy, due to the good performance of NEFEM with coarse meshes and high-order approximations.

Finally, as a more challenging application, the scattering by a PEC NASA almond of characteristic length 21λ is considered. The mesh employed for the computation has 48 699 elements with planar faces and 6 008 curved elements (3 504 with a face on the NURBS boundary, and 2 504 with an edge on the NURBS boundary).

Figure 3.55 (a) shows the surface triangular mesh of the almond and two components of the scattered field for a NEFEM solution with $p = 3$, for a wave incident onto the tip of the almond. The RCS distribution for vertical and horizontal polarization is represented in Figure 3.56.

The results compare well with published results by Hachemi et al. (2004), and again show the competitiveness of NEFEM for higher frequency problems. The tetrahedral mesh used by Hachemi et al. (2004) has 51 342 008 linear elements, and approximately 8 million of nodes. The surface mesh of the almond has 149 720 triangular elements. With NEFEM and $p = 3$ the mesh has 1 million of nodes (including the duplication due to the DG formulation). Even if a degree of approximation $p = 4$ is adopted to enhance the quality of the RCS pattern near viewing angles $-\pi$ and π, see Figure 3.56, the resulting mesh has 2 million of nodes, requiring fourth times less memory



(a) Vertical polarization (b) Horizontal polarization

Figure 3.56: Scattering by a PEC NASA almond of characteristic length 21λ: RCS for a NEFEM solution with $p = 3$

than using linear FEs.

Chapter 4

Summary and future developments

This thesis presents the development and application of a new FE technique for an efficient treatment of curved boundaries, the NURBS-enhanced finite element method (NEFEM). It considers the exact description of the geometry by means of its NURBS boundary representation, that is, the information usually provided by CAD software. At elements intersecting the NURBS boundary specific interpolation and numerical integration are proposed and, at elements not intersecting the boundary classical FE are used, preserving the efficiency of the FEM.

This chapter describes the main contributions of this thesis and details some research lines that open the introduction of this novel methodology.

4.1 Summary and contributions

The most relevant contributions of this thesis are briefly summarized next:

1. **Development of NEFEM.** NEFEM is proposed in Chapter 2, see also Sevilla et al. (2008a). Every *interior* element (i.e. elements not having an edge or face in contact with the NURBS boundary) can be defined and treated as standard FE or DG elements. Therefore, in the vast majority of the domain, interpolation and numerical integration are standard, preserving the computational efficiency of the classical FEM. Specific numerical strategies for interpolation and numerical

integration are needed only for those elements affected by the NURBS boundary representation.

NEFEM defines the approximation directly with cartesian coordinates. Different nodal distributions in curved elements are considered and the influence on the condition number is studied. For moderate degree of approximation, namely $p \leq 5$, it is sufficient to use distributions non adapted to the exact geometry. But, when very high-order approximations are employed, that is $p > 5$, the use of adapted nodal distributions imply an important improvement in the condition number of the element mass matrix.

Chapter 2 also proposes, discusses and compares different strategies to perform numerical integration, see also Sevilla and Fernández-Méndez (2009). For boundary integrals, several well known quadratures are compared, and the influence of the number of integration points in the accuracy of the computation is investigated. For interior integrals, Sevilla and Fernández-Méndez (2009) propose and compare two strategies. The most efficient one is recalled in Appendix B. When exact integration is feasible, explicit formulas for the selection of the number of integration points are deduced. Moreover, numerical examples are presented in order to show the influence of the number of integration points in the accuracy of the computation.

The extension of NEFEM to 3D domains is presented in detail in this thesis. Chapter 2 presents the core concepts of NEFEM in 3D, and Appendix B details all the technicalities. The quality of the approximation is analyzed, and the influence of the nodal distribution in the convergence rate is discussed. The strategy to perform the numerical integration is detailed in Chapter 2 and Appendix B. The key idea is to design specific quadratures for two element typologies, namely elements with only one face or one edge on the NURBS boundary. Any other element is split in order to obtain elements with only one face or edge on the NURBS boundary. This strategy avoids a special treatment of each element typology, and therefore, the implementation effort is considerably reduced. It is worth remarking that subdivisions are only applied to design

a numerical quadrature, thus no new degrees of freedom are introduced. More complex situations, involving trimmed and singular NURBS, boundary layers and curved material interfaces are also discussed.

NEFEM *a priori* error estimates for a second-order elliptic problem are given in Chapter 2, including estimates for h and p convergence. The proof is discussed and checked using numerical examples in Chapter 2. Further discussion and examples are presented in Appendix B, see also Sevilla et al. (2008a).

An important goal of the thesis is to provide an easy-to-implement integration of the CAD boundary representation in a FE code. Appendix B describes the little effort required to transform a standard FE code into a NEFEM code. An attractive feature of the proposed implementation is that the usual routines of a standard FE code can be directly used, namely routines for the computation of elemental matrices and vectors, assembly, etc. It is only necessary to implement new routines for the computation of numerical quadratures in curved elements and to store the value of the shape functions at integration points.

2. **Applications.** As pointed out in the introduction, electromagnetic scattering and compressible flow problems are two areas in which an accurate geometrical model is critical.

   Chapter 3 presents the application of NEFEM to electromagnetic scattering problems in 2D and 3D, using a DG formulation. Numerical examples include classical test cases, benchmarks for CEM codes, problems involving material interfaces, airfoils and with scatters of more complex geometry. The ability of NEFEM to compute an accurate solution with coarse meshes and high-order approximations is investigated. In particular, the possibilities of *NEFEM meshes*, with elements containing edge or corner singularities, are explored. The exact boundary representation allows to mesh the domain independently of the geometric complexity of the boundary, whereas standard FEs requires h-refinement to properly capture the geometry. With NEFEM the mesh size is no longer subsidiary to geometry complexity, and depends only on the accuracy require-

ments on the solution. Obviously, this drastic difference in mesh size results in a drastic memory savings, and also in computational cost if explicit time marching algorithms are considered. Thus, NEFEM is a powerful tool for scattering problems involving large-scale simulations with complex scatterers in 3D.

Another field in which a proper representation of the geometry is mandatory to obtain accurate results, is the DG solution of the Euler equations of gas dynamics. As pointed out by several authors, see Bassi and Rebay (1997), Van der Ven and Van der Vegt (2002) and Krivodonova and Berger (2006), the proper imposition of the outward unit normal in a solid wall boundary is crucial. In particular, if a linear approximation is used with DG, the spurious entropy production near curved walls prevents, in general, convergence to the correct solution, see Bassi and Rebay (1997). The application of NEFEM to compressible flow problems using a DG formulation is presented in Appendix F. With NEFEM, the exact imposition of the solid wall boundary condition provides accurate results even with a linear approximation of the solution. The exact boundary representation allows to use coarse meshes, but ensuring the proper implementation of the solid wall boundary condition. Further discussion on the application of NEFEM to Euler equations and examples are presented in Appendix F, corroborating the conclusions by Sevilla et al. (2008b).

3. **Review and comparison.** Several curved FE techniques have been proposed in the literature. Section 1.1 offers an overview, with a historical perspective.

   In this thesis NEFEM is compared with some popular curved FEs (namely isoparametric FEs, cartesian FEs and p-FEM), from three different perspectives: theoretical aspects, implementation and performance. Appendix C is devoted to compare these methodologies from a theoretical point of view, highlighting the main differences, and to describe the implementation issues of each methodology. Particular emphasis is placed in the numerical integration for p-FEM and NEFEM, with an exact boundary representation.

   The performance comparison is presented in Chapter 2 for second-order elliptic

problems, in Chapter 3 for electromagnetic scattering, and in Appendix F for compressible flow applications.

In every example shown, NEFEM is at least one order of magnitude more accurate compared to isoparametric FEs, for the same number of degrees of freedom. NEFEM is more accurate than cartesian FEs and also more accurate than p-FEM (with an exact boundary representation), showing that the cartesian approximation combined with an exact boundary representation, i.e. NEFEM, provides the maximum accuracy for a given spatial discretization. Moreover, for a desired accuracy NEFEM is also more computational efficient. In some examples, NEFEM only needs 50% of the number of degrees of freedom required by isoparametric FEs or p-FEM. In 3D, numerical examples show that methods with an approximate boundary representation (i.e. isoparametric FEs and cartesian FEs) may suffer an important loss of accuracy in coarse meshes, preventing, in some cases, optimal p-convergence. Whereas exact boundary representation considered by NEFEM ensures optimal convergence in all cases.

To conclude, the use of NEFEM is strongly recommended in the presence of curved boundaries and/or when the boundary of the domain has complex geometric details.

4. **Comparison of perfectly matched layers for scattering problems.** A key issue in the numerical solution of electromagnetic scattering problems is using a mechanism to perform the absorption of outgoing waves. A PML is considered in the examples in Chapter 3. Appendix E presents a comparison of two PMLs, the original PML of Berenger (1994), and a non-linear PML recently proposed by Abarbanel et al. (2006). First, a numerical study is presented in order to select the optimal parameters of each PML in a high-order DG framework. It is worth noting that the conclusions of previous analysis in a FD or FE context can not be directly extrapolated to high-order DG methods. For Berenger's PML, the optimal parameters are found to be different to those obtained in a FD or FE framework, and for the non-linear PML, no previous experience is

available. Finally, both PMLs are compared in terms of accuracy and some comments on memory requirements are given. For a desired accuracy, the non-linear PML is more efficient due to its ability to perform the absorption without extra (artificial) variables in the PML region.

4.2 Future developments

The potential of NEFEM for the treatment of curved boundaries is demonstrated in this work. However, several research lines are still open:

1. **Adaptivity.** In practical FE adaptive processes, see for instance Huerta et al. (1999), the computational mesh must be locally refined (or the polynomial order of the approximation increased) to properly approximate both the solution and the geometry. Whereas in a NEFEM context, the adaptive process is controlled only by the complexity of the solution, independently of the geometrical complexity of the domain, reducing therefore the necessary number of degrees of freedom to achieve a desired accuracy. Thus, the study of h and p adaptive processes in a NEFEM framework is worth to be investigated and compared with other FE techniques.

2. **Geometry singularities.** The potential of NEFEM meshes for high-order computations in coarse meshes has been shown in this work. NEFEM meshes offer a drastically reduction of the number of degrees of freedom compared to standard FE meshes. Nevertheless, one of the topics that deserves more attention is the performance of elements containing corner or edge singularities. Future work is needed to improve the quality of the approximation in the presence of singular solutions. The enrichment of the polynomial basis, well known for standard FEM meshes, see among others, Costabel and Dauge (1997), is worth to be investigated in the NEFEM context.

   NEFEM has exhibited optimal convergence when the singularity on the boundary does not imply a singularity in the solution. Therefore, the application of

NEFEM in this context, for instance to problems of swell propagation in coastal zones, is an interesting scenario for NEFEM.

3. **NEFEM extension.** NEFEM has been developed for tetrahedral and triangular meshes, but the ideas are readily extensible to other element typologies, i.e. hexahedrons, prisms, quadrilaterals, etc. In addition, the NEFEM concept may be exported to other numerical methodologies involving the numerical solution of partial differential equations in domains with curved boundaries. Of particular interest may be the application of the NEFEM technology to meshfree methods, see Fernández-Méndez (2001) and Belytschko et al. (2004). The flexibility to define the spatial discretization can be combined with a NEFEM coarse mesh to perform the numerical integration accounting for the exact geometry. The exact computation of the outward unit normal in NEFEM, can also be used for an accurate imposition of boundary conditions in a weak sense.

   Other numerical techniques, such as finite volumes or edge elements, may also take advantage of an exact boundary representation in a NEFEM fashion. Finally, it is also necessary to compare the existing curved FE techniques with NEFEM in other problems, in order to identify situations where the exact boundary representation considered in NEFEM can be advantageous.

4. **Mesh generation.** The generation of coarse meshes of complex geometric objects for NEFEM computations is not a trivial task. Although NEFEM does not need a high-order mesh generator, the use of a linear mesh generator may lead to non-valid meshes, due to the replacement of the exact boundary by the auxiliary boundary in the mesh generation process, see Roca (2009).

   Moreover, tools for meshing complex geometries without refinement near the details or singularities in the boundary, are not provided by standard mesh generators. Thus, automatic mesh generation technology for NEFEM is a new challenge by itself.

5. **Nodal distributions.** Specifically designed nodal distributions are needed in a continuous Galerkin framework to obtain optimal convergence with NEFEM.

This is formally stated in Chapter 2 and corroborated with the numerical examples in Appendix B.

The adaptation of nodal distributions, initially designed in elements with planar faces/edges, to the exact geometry of curved NEFEM elements provides accurate results in several situations. However, a general procedure to obtain optimal nodal distributions in 3D curved elements is desired.

6. **Numerical integration.** The proposed strategy to perform the numerical integration in NEFEM is based on parametrizations defined as a linear convex combination of an internal vertex and a boundary face/edge. For elements with several faces or edges on the NURBS boundary the strategy is based on subdivision, for an ease of implementation. However, more efficient strategies are desired for elements that can not be defined as a linear convex combination of an internal vertex and a boundary face/edge, see comments in Section B.2.

Appendix A

Non-Uniform Rational B-Splines (NURBS)

This chapter is devoted to introduce some basic concepts on NURBS curves and surfaces. Excellent references are available with presentations of NURBS, see for instance Piegl and Tiller (1995).

A.1 NURBS curves

A qth-degree NURBS curve is a piecewise rational function defined in parametric form as

$$\boldsymbol{C}(\lambda) = \left(\sum_{i=0}^{\mathtt{n_{cp}}} \nu_i \, \boldsymbol{B}_i \, C_i^q(\lambda)\right) \Big/ \left(\sum_{i=0}^{\mathtt{n_{cp}}} \nu_i \, C_i^q(\lambda)\right) \qquad 0 \leq \lambda \leq 1,$$

where $\{\boldsymbol{B}_i\}$ are the coordinates of the $\mathtt{n_{cp}} + 1$ *control points* (forming the *control polygon*), $\{\nu_i\}$ are the control weights, and $\{C_i^q(\lambda)\}$ are the normalized B-spline basis functions of degree q, which are defined recursively by

$$C_i^0(\lambda) = \begin{cases} 1 & \text{if } \lambda \in [\lambda_i, \lambda_{i+1}[, \\ 0 & \text{elsewhere,} \end{cases}$$

$$C_i^k(\lambda) = \frac{\lambda - \lambda_i}{\lambda_{i+k} - \lambda_i} C_i^{k-1}(\lambda) + \frac{\lambda_{i+k+1} - \lambda}{\lambda_{i+k+1} - \lambda_{i+1}} C_{i+1}^{k-1}(\lambda),$$

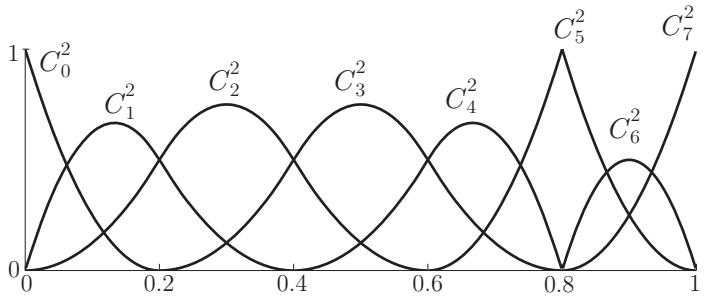


Figure A.1: B-spline basis functions for the knot vector (A.1)

for $k = 1 \dots q$, where λ_i, for $i = 0, \dots, \mathtt{n_k}$, are the *knots* or *breakpoints*, which are assumed ordered $0 \leq \lambda_i \leq \lambda_{i+1} \leq 1$. They form the so-called *knot vector*

$$\Lambda = \{\underbrace{0, \dots, 0}_{q+1}, \lambda_{q+1}, \dots, \lambda_{\mathtt{n_k}-q-1}, \underbrace{1, \dots, 1}_{q+1}\},$$

which uniquely describes the B-spline basis functions. The multiplicity of a knot, when it is larger than one, determines the decrease in the number of continuous derivatives. Control points, $\mathtt{n_{cp}}$+1, and knots, $\mathtt{n_k}$+1, are related to the degree of the parametrization, q, by the relation $\mathtt{n_k} = \mathtt{n_{cp}} + q + 1$, see Piegl and Tiller (1995) for more details. Figure A.1 shows the B-spline basis functions for the knot vector

$$\Lambda = \{0, 0, 0, 0.2, 0.4, 0.6, 0.8, 0.8, 1, 1, 1\}. \tag{A.1}$$

Note that NURBS are piecewise rational functions, whose definition changes at knots. An example of a NURBS curve is represented in Figure A.2 with the corresponding control polygon. The image of the breakpoints or knots by the NURBS are depicted in order to stress the discontinuous definition of the parametrization. In practice CAD manipulators work with *trimmed* NURBS, which are defined as the initial parametrization restricted to a subspace of the parametric space. Figure A.3 shows the NURBS curve of Figure A.2 trimmed to the subinterval $[0.05, 0.75]$.

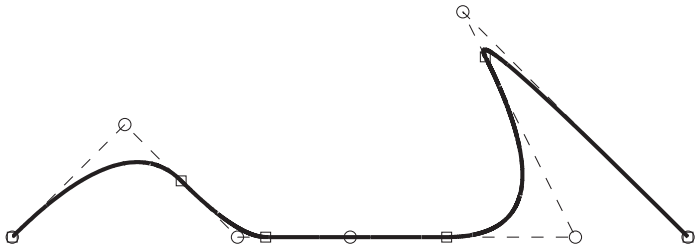


Figure A.2: NURBS curve (solid line), control points (denoted by ∘), control polygon (dashed line) and breakpoints (denoted by □)

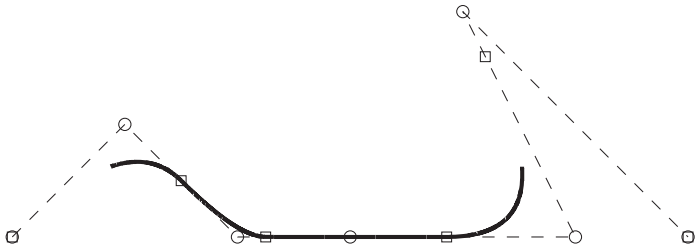


Figure A.3: Trimmed NURBS curve with $\lambda \in [0.05, 0.75]$ (solid line), control points (denoted by ∘), control polygon (dashed line) and breakpoints (denoted by □)

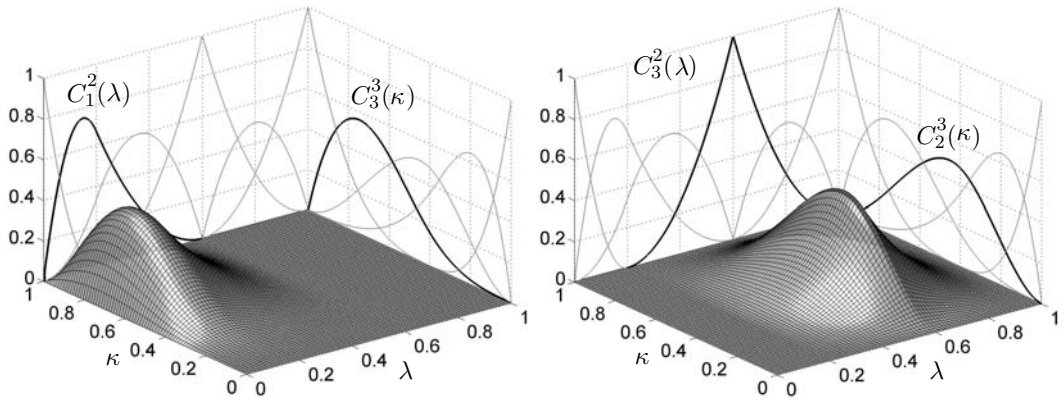
A.2 NURBS surfaces

A NURBS surface of degree q in λ and degree l in κ, is a piecewise rational function defined in parametric form as

$$\boldsymbol{S}(\lambda,\kappa) = \left(\sum_{i=0}^{\mathtt{n}_{\mathtt{cp}}^{\lambda}}\sum_{j=0}^{\mathtt{n}_{\mathtt{cp}}^{\kappa}} \nu_{ij}\,\boldsymbol{B}_{ij}\,S_{i,j}^{q,l}(\lambda,\kappa)\right) \Big/ \left(\sum_{i=0}^{\mathtt{n}_{\mathtt{cp}}^{\lambda}}\sum_{j=0}^{\mathtt{n}_{\mathtt{cp}}^{\kappa}} \nu_{ij}\,S_{i,j}^{q,l}(\lambda,\kappa)\right), \quad 0 \le \lambda,\kappa \le 1,$$

where $\{\boldsymbol{B}_{ij}\}$ are the coordinates of the $(\mathtt{n}_{\mathtt{cp}}^{\lambda}+1)(\mathtt{n}_{\mathtt{cp}}^{\kappa}+1)$ *control points* (defining the *control net*), $\{\nu_{ij}\}$ are the control weights, and $\{S_{i,j}^{q,l}(\lambda,\kappa)\}$ are the 2D B-spline basis functions of degree q in λ and l in κ. Each 2D B-Spline basis function is defined as a tensor product of 1D basis functions, that is

$$S_{i,j}^{q,l}(\lambda,\kappa) := C_i^q(\lambda)C_j^l(\kappa). \tag{A.2}$$



(a) $S_{1,3}^{2,3}(\lambda,\kappa) := C_1^2(\lambda)C_3^3(\kappa)$ (b) $S_{3,2}^{2,3}(\lambda,\kappa) := C_3^2(\lambda)C_2^3(\kappa)$

Figure A.4: Example of 2D B-spline basis functions

Figure A.4 shows two 2D B-spline basis functions for knot vectors

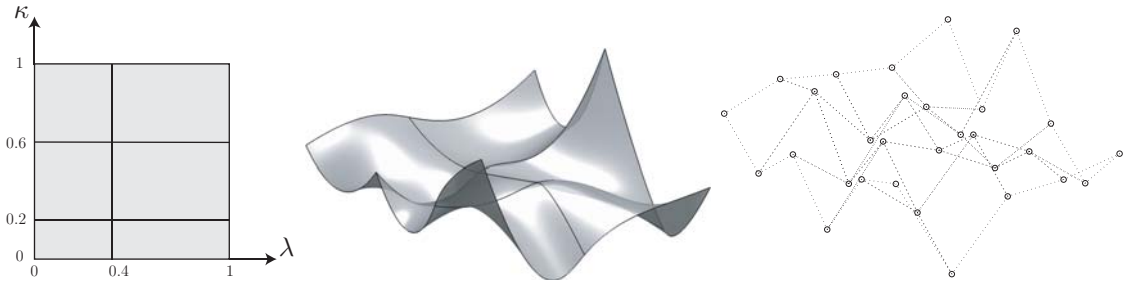
$$\Lambda^\lambda = \{0,0,0,0.2,0.6,0.6,1,1,1\},$$

$$\Lambda^\kappa = \{0,0,0,0,0.4,1,1,1,1\}.$$

Complete 1D basis are represented for each direction to illustrate the construction of 2D basis functions (A.2).

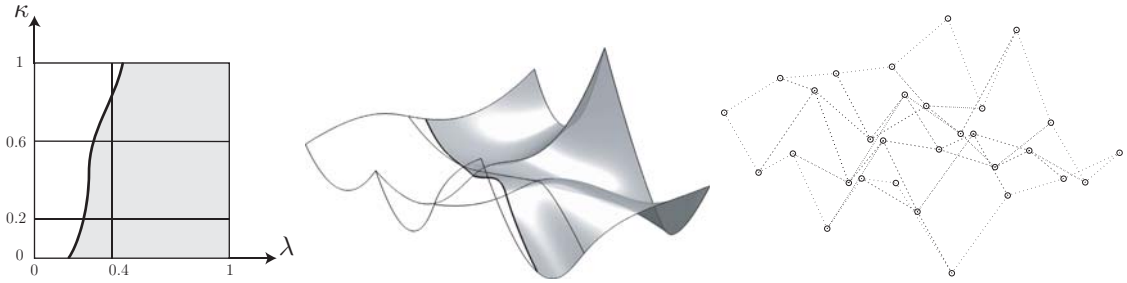
Note that NURBS surfaces change their definition along *knot lines*, that is when $\lambda = \lambda_i$, for $i = 1,\ldots,\mathtt{n}_\mathtt{k}^\lambda$, or $\kappa = \kappa_i$, for $i = 1,\ldots,\mathtt{n}_\mathtt{k}^\kappa$. An example of a NURBS surface is represented in Figure A.5 with the corresponding control net. Knot lines are represented on the NURBS surface in order to stress the discontinuous nature of the parametrization. An example of a trimmed NURBS surface is represented in Figure A.6, showing the NURBS surface of Figure A.5 trimmed with the thick curve. In practical applications, it is also common to deal with *singular* (or *singularly parametrized*) NURBS surfaces. Such surfaces contain at least one *singular point*, defined as a point where a directional derivative is zero. For these surfaces, knot lines typically converge to the singular point, see an example in Figure A.7.

Singular and trimmed NURBS are often present in real CAD models. For instance, the complete aircraft represented in Figure A.8 is composed of several trimmed



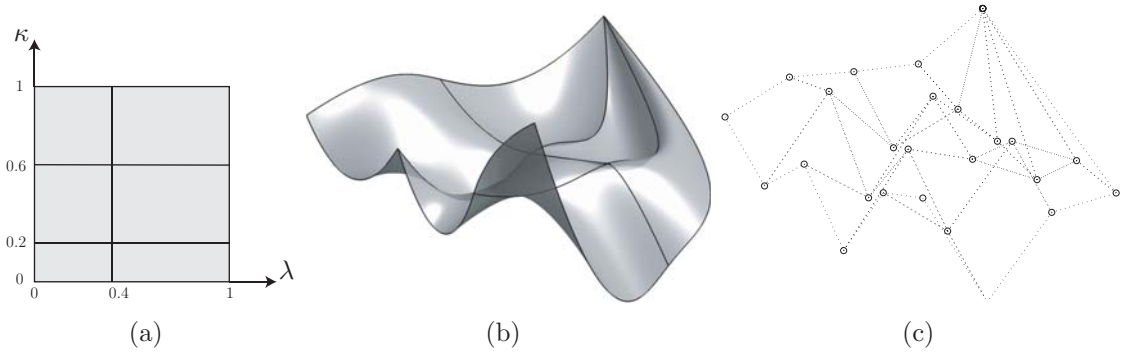
(a) (b) (c)

Figure A.5: (a) Parametric space, (b) NURBS surface with knot lines, and (c) control net



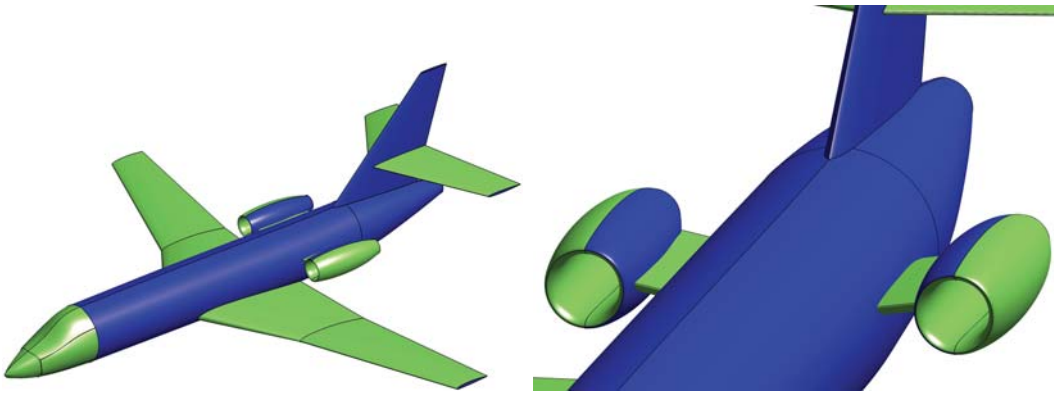
(a) (b) (c)

Figure A.6: (a) Parametric space trimmed by the thick curve, (b) trimmed NURBS surface with knot lines and the thick curve used to trim the initial surface of Figure A.5, and (c) control net



(a) (b) (c)

Figure A.7: (a) Parametric space, (b) singular NURBS surface with knot lines, and (c) control net



(a) (b)

Figure A.8: (a) NURBS surfaces corresponding to a complete aircraft, and (b) detail near the engine air intakes

NURBS surfaces (in blue). As it will be shown in the following appendix, NEFEM is able to deal with such surfaces with trivial modifications in the design of numerical quadratures for elements affected by the trimmed curve in the parametric space. This is another advantage with respect to the isogeometric analysis proposed by Hughes et al. (2005), where trimmed NURBS surfaces can not be considered.

Appendix B

Getting to the heart of NEFEM

This Appendix describes technical details of NEFEM that were omitted for clarity in Chapter 2. Further analysis on the polynomial basis adopted in NEFEM is presented. Particular emphasis is placed on the influence of the nodal distribution on the convergence rates. Main results of the numerical integration in the 2D case are summarized and the details of the 3D case are discussed. Finally, some comments on the implementation of NEFEM are given, and the pre and postprocess stages in a NEFEM computation are described.

B.1 Polynomial interpolation

NEFEM considers the polynomial basis in cartesian coordinates, as described in Section 2.2. To make the computation of Lagrange polynomial basis more systematic, for any degree and for any distribution of nodes, the implementation proposed by Hesthaven and Warburton (2002) is adopted. A polynomial basis $\{P_i(\boldsymbol{x})\}$, with the required degree and whose definition is independent of the nodal coordinates, is considered. Then, given a nodal distribution in Ω_e, with coordinates $\{\boldsymbol{x}_i\}_{i=1}^{\mathtt{n}_{\mathtt{en}}}$, the Lagrange polynomial basis $\{N_i(\boldsymbol{x})\}_{i=1}^{\mathtt{n}_{\mathtt{en}}}$ can be expressed in terms of the polynomial basis $\{P_i(\boldsymbol{x})\}_{i=1}^{\mathtt{n}_{\mathtt{en}}}$ as

$$N_i(\boldsymbol{x}) = \sum_{j=1}^{\mathtt{n}_{\mathtt{en}}} \left[\boldsymbol{V}^{-1}\right]_{ji} P_j(\boldsymbol{x}), \tag{B.1}$$

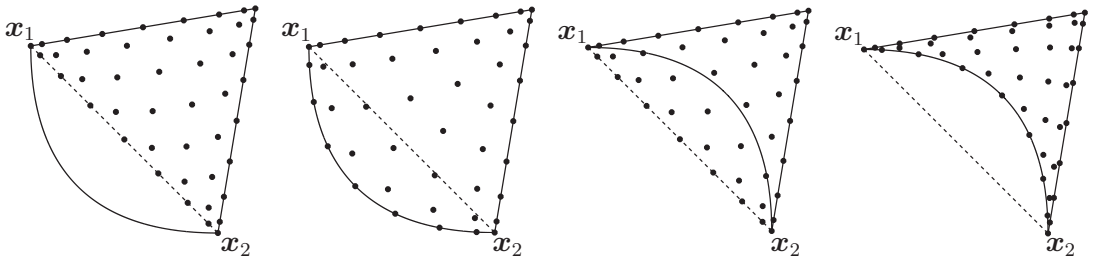


Figure B.1: Nodal distribution on curved triangular elements with a NURBS edge

where $\mathtt{n_{en}}$ is the number of element nodes and the multidimensional Vandermonde matrix is defined as $V_{ij} := P_j(\boldsymbol{x}_i)$, for $i, j = 1, \ldots, \mathtt{n_{en}}$. Note that Equation (B.1) holds for any polynomial basis $\{P_i(\boldsymbol{x})\}_{i=1}^{\mathtt{n_{en}}}$. Here an orthogonal polynomial basis $\{P_i(\boldsymbol{x})\}_{i=1}^{\mathtt{n_{en}}}$ derived from the Jacobi polynomials is considered, to ensure moderate condition number for the Vandermonde matrix $\boldsymbol{V}$, see Hesthaven and Warburton (2002) and references therein. Moreover, orthogonal polynomial basis allows analytical evaluation of some inner products in straight-sided elements (Szegö, 1975).

Different options can be considered for the definition of a nodal distribution in Ω_e. The influence of the nodal distribution on the condition number of the elemental mass matrix is studied in Section 2.2. A similar analysis is performed by Sevilla et al. (2008a) for the 2D case, showing the influence of Fekette nodal distributions on the condition number of the elemental matrices. Here the influence of different NURBS boundaries on the condition number of the elemental mass matrix is explored and illustrated with 2D and 3D examples. Finally, a relevant characteristic of the cartesian approximation adopted in NEFEM is discussed and analyzed. This characteristic motivates the additional hypothesis of Fekette nodal distributions in Theorem 1 to guarantee optimal convergence with a strong imposition of Dirichlet boundary conditions.

Let Ω_e be an element with a curved edge containing the vertices $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, and $\boldsymbol{C}$ its NURBS parametrization. Figure B.1 represents different nodal distributions in convex and non-convex curved elements. Fekette nodal distributions on the straight-sided triangle given by the vertices of Ω_e, see Taylor et al. (2000), or adapted to the exact geometry, are considered. Figure B.2 shows the condition number of the mass

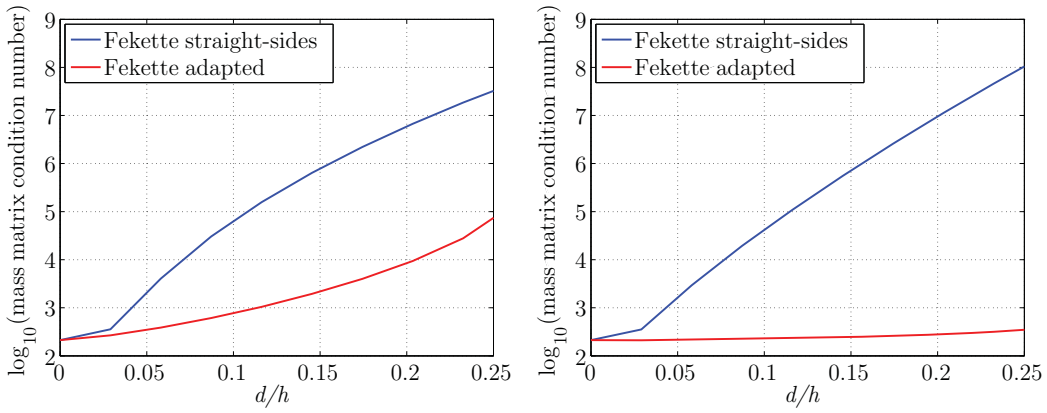


Figure B.2: Condition number of the mass matrix as a function of d/h for a (a) convex element, and for a (b) non-convex element

matrix as a function of d/h, where h is the element size and d is the maximum distance between the curved boundary and the straight line connecting $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$.

When the nodal distribution is not adapted to the exact geometry, an increase in the maximum distance d implies an important growth in the condition number of the mass matrix. However, for adapted distributions, different behavior is observed. In fact, for large values of d/h the use of adapted distributions is mandatory in order to avoid the ill-conditioning of the elemental matrices.

In most situations, internal faces/edges are straight, but there are a number of situations in which internal curved faces/edges are present in a computational mesh. For instance, curved internal faces/edges must be considered in a boundary layer or in the presence of a curved material interface. Moreover, in 3D, internal faces with an edge on the NURBS boundary are curved, see an example in Figure B.3. For boundary layers, internal curved faces/edges can be described using polynomial functions, for example by using an isoparametric mapping. For curved interfaces the situation is more complex, as the exact geometry of the interface must be considered.

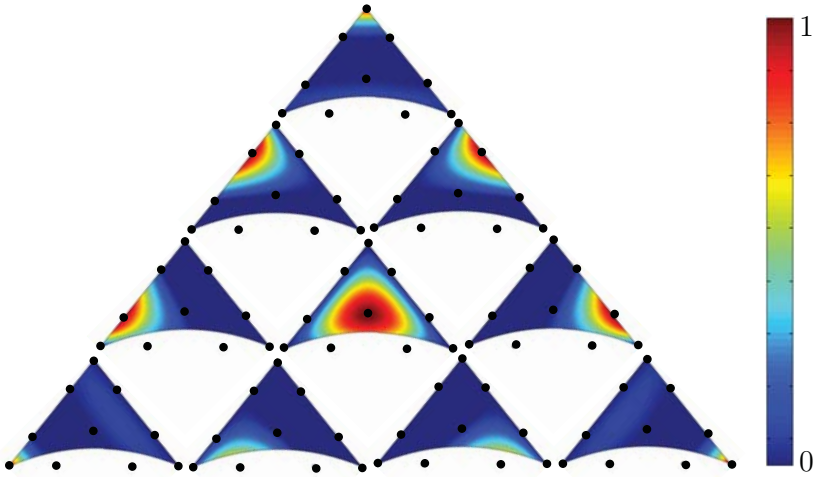
Let us consider a triangular element Ω_e with a curved edge, Γ_e, and $\{N_i(\boldsymbol{x})\}_{i=1}^{\mathtt{n_{en}}}$ its corresponding polynomial basis functions. For isoparametric FEs, all shape functions associated to nodes not in the curved edge Γ_e vanish along this edge. This means

Figure B.3: Mesh of a sphere showing some internal curved faces

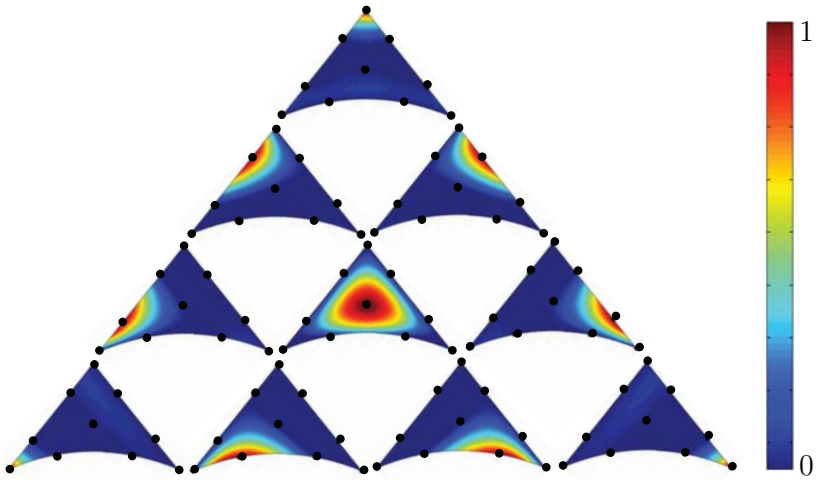
that the interpolation in an element edge is defined only in terms of basis functions associated to nodes in that edge, ensuring the continuity of the solution across internal edges. If a cartesian approximation is considered, i.e. cartesian FEs or NEFEM, the situation is different because all shape functions do not vanish along the curved edge, see an example of the polynomial basis functions for a NEFEM element with a degree of approximation $p = 3$ in Figure B.4. Thus, even if a standard Galerkin formulation is considered, the continuity of the solution across internal curved edges is not guaranteed, and the convergence properties of the approximation may be affected. Obviously, this is not the case for a DG approximation. In a DG framework, the continuity of the solution between elements is weakly imposed, with numerical fluxes. Therefore, optimal convergence rates are retained with no dependence on the nodal distribution. To check the optimal convergence in a DG framework, let us consider the numerical solution of the scalar convection equation

$$\left\{ \begin{array}{rcl} \dfrac{\partial u}{\partial t} + a_k \dfrac{\partial u}{\partial x_k} & = & 0 \quad \text{in} \quad \Omega \\ u & = & u_d \quad \text{on} \quad \Gamma_d \end{array} \right.$$

where $\boldsymbol{a} = (a_1, a_2, a_3)$ is the convection velocity and the domain Ω is a cylinder, see a coarse mesh with internal curved faces in Figure B.5 (a). A Dirichlet boundary condition is imposed in the lower part of the cylinder, corresponding to a sinusoidal wave that enters in the cylinder and propagates in the x_3^+ direction, i.e. $\mathbf{a} = (0, 0, 1)$.

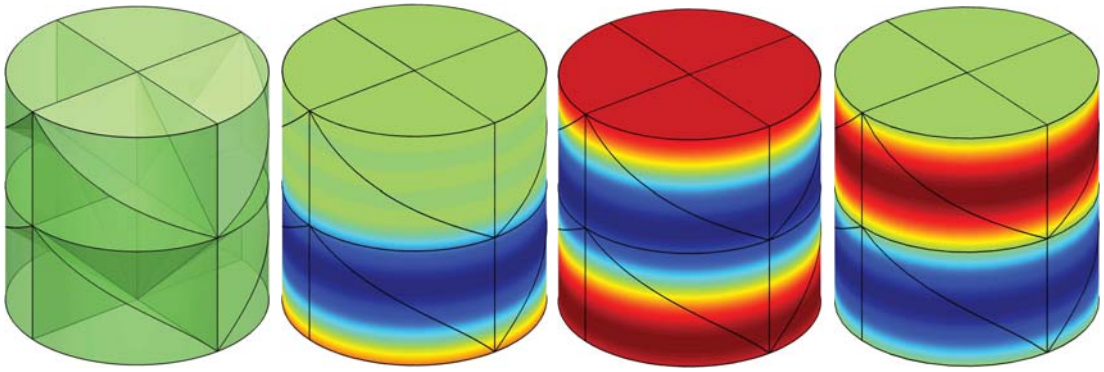


(a)



(b)

Figure B.4: Polynomial basis functions for a NEFEM curved element with Fekette nodal distributions for $p = 3$: (a) on the straight-sided triangle given by its vertices, and (b) adapted to the exact geometry



(a) Mesh (b) $t = 1.2$ (c) $t = 3.8$ (d) $t = 6.0$

Figure B.5: Convection in a cylinder: (a) coarse mesh of a cylinder with 24 curved tetrahedrons and snapshots of the numerical solution computed with NEFEM and $p = 8$ in (b), (c) and (d)

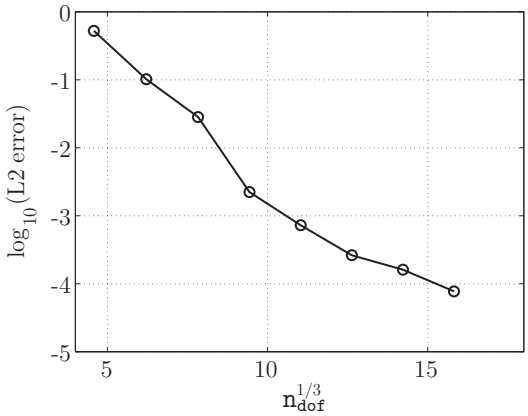
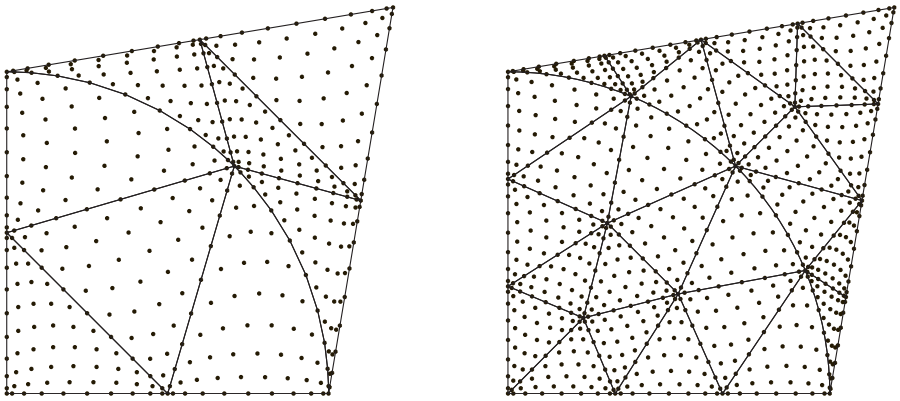


Figure B.6: Convection in a cylinder: convergence of the error in the $\mathcal{L}^2(\Omega)$ norm for increasing p

The numerical solution for a degree of interpolation $p = 8$ is also represented in Figure B.5 at different times. To check the optimal rate of convergence, the evolution of the error in the $\mathcal{L}^2(\Omega)$ norm is represented in Figure B.6 as p is uniformly increased from $p = 1$ up to $p = 8$. For each element, Ω_e, equally-spaced nodal distributions on the tetrahedral with planar faces given by the vertices of Ω_e are considered, and the optimal rate of convergence is achieved.

To illustrate the behavior of NEFEM in a standard Galerkin framework, the fol-



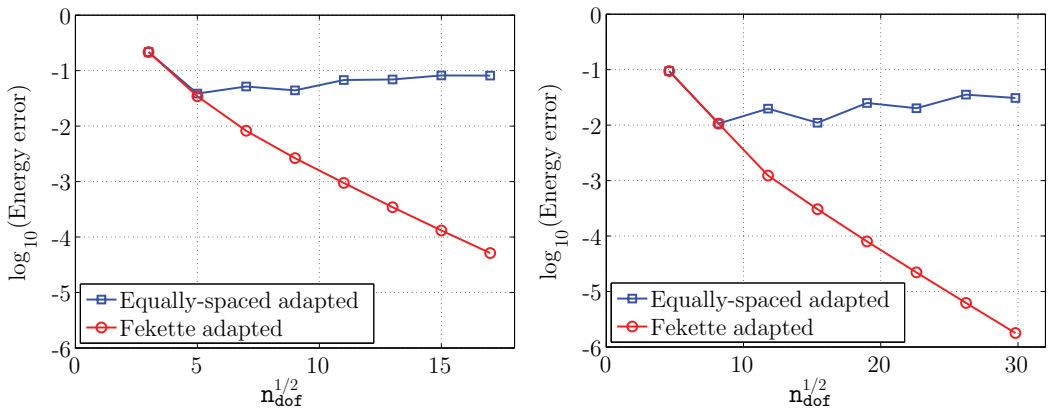
(a) Mesh 1 (b) Mesh 2

Figure B.7: Two triangular meshes with interior curved edges and Fekette nodal distributions adapted to curved elements for a degree of approximation $p = 8$

lowing Poisson problem is considered in 2D

$$\begin{cases} -\Delta u &= s \quad \text{in} \quad \Omega \\ u &= u_d \quad \text{on} \quad \partial\Omega \end{cases} \tag{B.2}$$

where Ω is the domain, see two computational meshes in Figure B.7. The analytical solution is $u(x, y) = x\cos(y) + y\sin(x)$, and the source term s is determined by analytical differentiation of u. Note that internal curved edges are present in this mesh, and therefore, specific nodal distributions are needed to achieve optimal performance, such as Fekette nodes. Figure B.8 shows a p-convergence comparison by using different nodal distributions in the discretization shown in Figure B.7. The error in energy norm is represented as a function of the square root of the number of degrees of freedom. When equally spaced nodal distributions are employed, exponential convergence is not achieved. In fact, the use of equally spaced nodes leads to an incorrect solution as the degree of interpolation is increased. The accuracy is substantially improved with Fekette nodal distributions adapted to the exact geometry, showing the expected (exponential) convergence. It is important to remark that, in general, NURBS are not parametrized by the arc length parameter. Therefore, a Fekette nodal distribution on the physical space is not obtained as the image of a Fekette nodal distribution on the



(a) Mesh 1 (b) Mesh 2

Figure B.8: p-convergence of the error in the energy norm for different nodal distributions on the meshes of Figure B.7

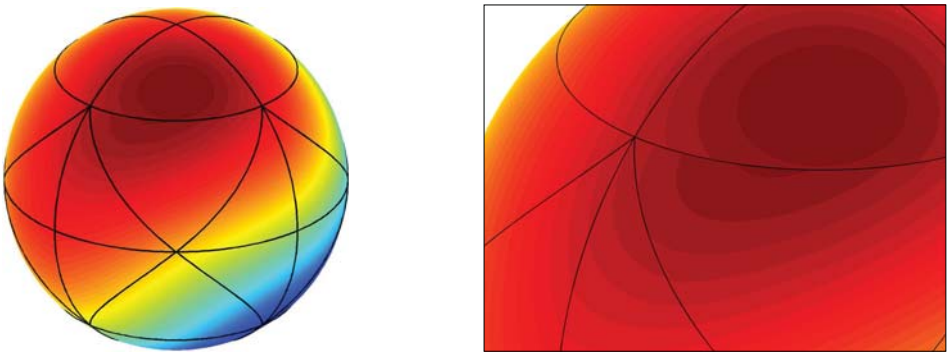


Figure B.9: Solution of the Poisson problem in a sphere and detail of the solution showing a small discontinuity across curved boundary edges

parametric space of the NURBS.

A similar performance is observed in 3D simulations, see an example in Figure B.9. The Poisson problem (B.2) in solved in a sphere with the computational mesh represented in Figure B.3. A detailed view of the numerical solution reveals a small discontinuity of the solution across curved boundary edges.

B.2 Numerical integration

This section describes in detail the strategy to perform the numerical integration in NEFEM, for both 2D and 3D domains. First, the key ideas of the numerical integration for NEFEM in 2D domains are recalled, see Sevilla and Fernández-Méndez (2009) for a complete presentation, and a discussion of more general situations is presented. Finally, for 3D domains, this section supplements the brief presentation in Section 2.3. A detailed description of the strategy to perform the numerical integration in elements affected by the NURBS boundary representation is given. More complex situations involving trimmed and singular NURBS are also considered.

B.2.1 Two dimensional case

In Sevilla and Fernández-Méndez (2009) a detailed comparison and discussion on different alternatives to evaluate the integrals of polynomial functions along NURBS curves is presented. Numerical experiments reveal that Gauss-Legendre quadratures are a competitive choice in front of other quadrature rules such as trapezoidal and Simpson composite rules or Romberg's integration. Although the faster convergence is obviously obtained for high-order simple quadratures (in each patch), the use of composite rules is very attractive, because it allows the definition of adaptive quadratures to control the integration error and ensure reliable computations for any NURBS and any order of polynomial interpolation.

The computation of integrals in elements with a curved edge on the NURBS boundary is also analyzed by Sevilla and Fernández-Méndez (2009). The best alternative is to define a transformation from the rectangle $R = [\lambda_1^e, \lambda_2^e] \times [0, 1]$ to the curved element Ω_e, namely,

$$\begin{aligned} \boldsymbol{\psi} \ : R = [\lambda_1^e, \lambda_2^e] \times [0,1] &\longrightarrow \Omega_e \\ \boldsymbol{\lambda} = (\lambda, \vartheta) &\longmapsto \boldsymbol{\psi}(\boldsymbol{\lambda}) := \boldsymbol{C}(\lambda)(1-\vartheta) + \vartheta \boldsymbol{x}_3, \end{aligned} \tag{B.3}$$

where $\boldsymbol{x}_1 = \boldsymbol{C}(\lambda_1^e)$ and $\boldsymbol{x}_2 = \boldsymbol{C}(\lambda_2^e)$ are the vertices on the NURBS boundary, and

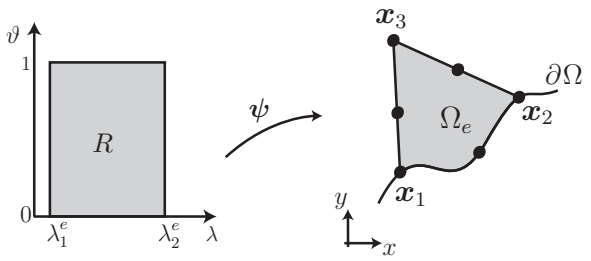


Figure B.10: Transformation from $[\lambda_1^e, \lambda_2^e] \times [0, 1]$ to a curved triangle Ω_e

$\boldsymbol{x}_3$ is the interior vertex of the element Ω_e, see Figure B.10. Then, the integral on a curved element is computed as

$$\int_{\Omega_e} f \, d\Omega = \int_R f\big(\boldsymbol{\psi}(\lambda, \vartheta)\big) \, \|J_{\boldsymbol{\psi}}(\lambda, \vartheta)\| \, d\lambda \, d\vartheta.$$

The integral can be evaluated using 1D Gauss-Legendre quadratures in each direction. Recall that application $\boldsymbol{\psi}$ is linear in the second parameter, ϑ. Therefore, integrals involved in the elemental matrices, for a NEFEM solution with interpolation of degree p, can be exactly computed for this direction, using a Gauss-Legendre quadrature with $p + 1$ integration points. If changes of NURBS definition are present in the curved edge, composite quadratures are considered in the first parameter, λ.

Another obvious option instead of using $\boldsymbol{\psi}$ to transform a rectangle into Ω_e, is to define another transformation from a triangle with straight sides to Ω_e and then use quadratures specifically designed for triangles. This is also discussed in Sevilla and Fernández-Méndez (2009). For standard FE these triangle quadratures require less integration points than other quadrature rules to achieve the same accuracy, but this is not the case here. The use of a transformation depending on the NURBS parametrization (from a straight-sided triangle to a curved triangle) leads to expensive triangle quadratures. The integration strategy proposed in this section is much more competitive due to the good behavior of parameter ϑ. The efficiency of the proposed quadrature is illustrated in Figure B.11. It shows the integration points required to integrate x over a curved triangle with an error of 0.5%, using the transformation from a rectangle proposed in this section (with 30 integration points) and a symmetric

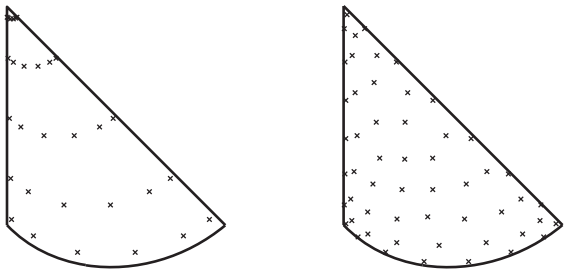


Figure B.11: Two numerical quadratures in a curved triangle for the same accuracy

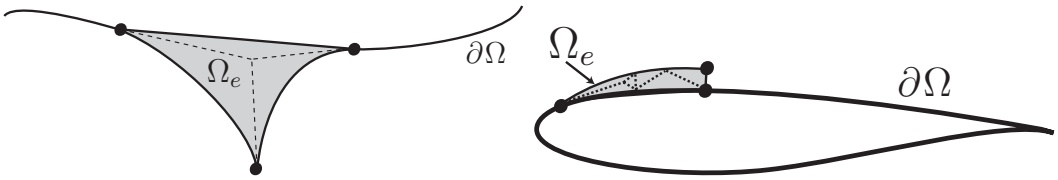


Figure B.12: Subdivision of a triangular element with several curved edges

triangle quadrature Wandzura and Xiao (2003) adapted to the curved element (with 54 integration points), see Sevilla and Fernández-Méndez (2009) for further details.

The definition of $\boldsymbol{\psi}$ in Equation B.3 assumes that the curved element Ω_e has one curved edge. Nevertheless, sometimes it is necessary to consider elements with two (or more) curved edges. For instance, Figure B.12 shows two possible situations: an element near a boundary corner and an element in a boundary layer. A simple strategy to design a numerical quadrature on an element with two or more curved edges is to split the element in order to obtain *subelements* with at most one curved edge. Then, a composite quadrature on Ω_e can be easily obtained with the strategy described in this section applied to subelements. It is important to recall that the subdivision is only applied to obtain a numerical quadrature on the curved element, avoiding the necessity of new parametrization for these special elements. Thus, no new degrees of freedom are introduced due to the element splitting.

Finally, note that application $\boldsymbol{\psi}$ is defined as a linear convex combination of a parametric curve and the interior vertex of Ω_e. Consequently, it is implicitly assumed that the straight line connecting the interior vertex and a given point of the parametric curve lies inside the element. The use of coarse meshes may lead to elements violating

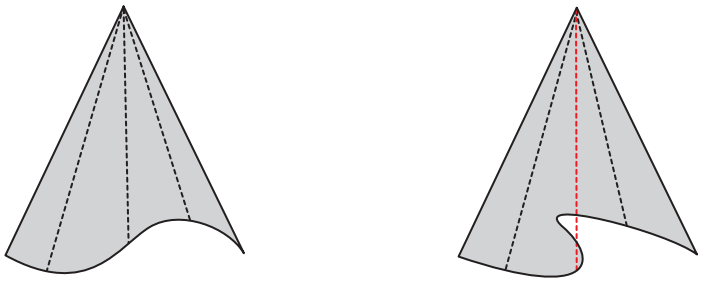


Figure B.13: Two curved triangular elements where discontinuous lines represent the parametrization $\boldsymbol{\psi}$

this property and therefore, it is necessary to develop efficient strategies to design numerical quadratures in such elements. Two examples are represented in in Figure B.13, showing an element that can be parametrized using the application $\boldsymbol{\psi}$ and an element that needs a special treatment. It is worth remarking that this problem is not exclusive of NEFEM meshes. Blending mappings used in p-FEM also suffers from this problem, and the mesh generation for high-order methods is not a trivial task, see a brief discussion in B.4.1 and Luo et al. (2002) for further details.

B.2.2 Three dimensional case

Section 2.3 describes the strategy to design numerical quadratures on tetrahedral elements with a curved face on the NURBS boundary. Obviously other element typologies must be considered in a NEFEM tetrahedral mesh. This section describes the strategy to perform the numerical integration in elements with one edge on the NURBS boundary, and, in general, in elements with several edges and/or faces on the NURBS boundary.

Let us consider an element Ω_e with one edge, Γ_e, on the NURBS boundary, see Figure B.14. This element has two internal curved faces corresponding to each one of the interior nodes $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$ and sharing the NURBS edge Γ_e. A parametrization for

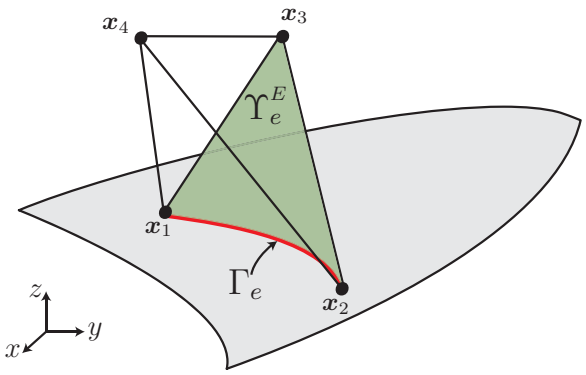


Figure B.14: Definition of a curved tetrahedral face, Υ_e^E, with an edge, Γ_e, on the NURBS boundary

a curved face corresponding to the interior vertex $\widehat{\boldsymbol{x}}$ is

$$\begin{aligned} \boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}} \; &: [0,1]^2 \longrightarrow \Upsilon_e^E \\ &(\varrho,\sigma) \longmapsto \boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}(\varrho,\sigma) := (1-\sigma)\boldsymbol{\theta}(\varrho) + \sigma\widehat{\boldsymbol{x}}, \end{aligned} \tag{B.4}$$

where $\boldsymbol{\theta}$ is the parametrization of the curved edge Γ_e. An integral on the curved face Υ_e^E can be written as

$$\int_{\Upsilon_e^E} f \; dx\, dy\, dz = \int_0^1 \int_0^1 f\big(\boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}(\varrho,\sigma)\big) \; \|J_{\boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}}(\varrho,\sigma)\| \; d\varrho \; d\sigma,$$

where f is a generic function, and $\|J_{\boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}}(\varrho,\sigma)\|$ denotes the norm of the differential of the mapping $\boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}$ (which, in general, is not a polynomial). Numerical integration can be performed using 1D Gauss-Legendre quadratures in each direction. In fact, application $\boldsymbol{\Theta}_{\widehat{\boldsymbol{x}}}$ is linear in the second parameter, σ, and exact integration is feasible in this direction. For a NEFEM solution with a degree of approximation p, the integral can be exactly computed for this direction, using a Gauss-Legendre quadrature with $p+1$ integration points. The numerical integration for the first direction, given by the NURBS parameter ϱ, presents the same difficulty as the integration over a NURBS curve, which is discussed in Sevilla and Fernández-Méndez (2009). As usual, the evaluation of the previous integral requires taking into account the discontinuous nature

of the NURBS parametrization, considering composite quadratures for ϱ direction.

To perform the numerical integration in Ω_e, the element is parametrized by

$$\begin{aligned} \boldsymbol{\Phi} \ &: [0,1]^3 \longrightarrow \Omega_e \\ &(\varrho,\sigma,\tau) \longmapsto \boldsymbol{\Phi}(\varrho,\sigma,\tau) := (1-\tau)\boldsymbol{\Theta}_{\boldsymbol{x_3}}(\varrho,\sigma) + \tau \boldsymbol{x}_4, \end{aligned} \tag{B.5}$$

where $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$ are the interior vertices of Ω_e. Note that the definition of parametrization $\boldsymbol{\Phi}$ in Equation (B.5) is independent on the order of the interior vertices $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$. That is, the element Ω_e can be equivalently parametrized by

$$\boldsymbol{\Phi}(\varrho,\sigma,\tau) := (1-\tau)\boldsymbol{\Theta}_{\boldsymbol{x_4}}(\varrho,\sigma) + \tau \boldsymbol{x}_3.$$

Then, element integrals can be written as

$$\int_{\Omega_e} f \, dx\, dy\, dz = \int_0^1 \int_0^1 \int_0^1 f\big(\boldsymbol{\Phi}(\varrho,\sigma,\tau)\big) \, |J_{\boldsymbol{\Phi}}(\varrho,\sigma,\tau)| \, d\varrho \, d\sigma \, d\tau,$$

where $|J_{\boldsymbol{\Phi}}|$ is the determinant of Jacobian of the transformation $\boldsymbol{\Phi}$. Note that application $\boldsymbol{\Phi}$ is linear in the second and third parameters, σ and τ. Therefore, integrals involved in the elemental matrices, for a NEFEM solution with interpolation of degree p, can be exactly computed for these directions, using a Gauss-Legendre quadrature with $p+2$ integration points. No exact integration is feasible in the NURBS direction ϱ, and composite quadratures must be considered if changes of NURBS definition are present.

In general, an element in a NEFEM computational mesh may have several edges and/or faces on the NURBS boundary. Nevertheless, to reduce the casuistics, such elements are split in elements with only one face or one edge on the NURBS boundary. It is worth remarking that subdivisions are only applied to design a numerical quadrature, no new degrees of freedom are introduced. This strategy avoids a special treatment of each element typology.

Two examples are presented to illustrate the proposed strategy. The first example

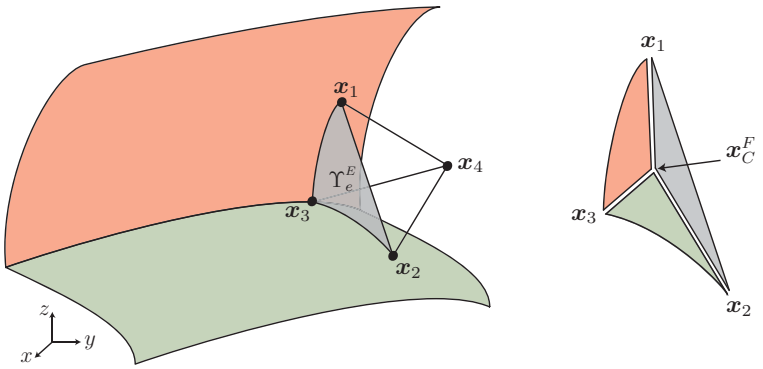


Figure B.15: Thetrahedral element with two edges defined by different NURBS surfaces, and subdivision of the tetrahedral curved face with two edges defined by different NURBS surfaces

considers a tetrahedral element with two edges defined by different NURBS surfaces, see Figure B.15. The curved face with two edges on the NURBS boundary, Υ_e^E, is split in three *subfaces* using its center of mass, $\boldsymbol{x}_C^F$, as represented in Figure B.15. Subfaces are defined as a linear convex combination of the edges of the initial face Υ_e^E and its center of mass $\boldsymbol{x}_C^F$. Therefore, the resulting subfaces have at most one edge on the NURBS boundary. In fact, after subdivision, two subfaces have one edge defined by a NURBS, and a numerical quadrature can be designed using the parametrization (B.4). The third face, given by $\boldsymbol{x}_2$, $\boldsymbol{x}_1$ and $\boldsymbol{x}_C^F$, is planar and a standard triangle quadrature can be implemented. To design a numerical quadrature on the curved element Ω_e three *subelements* are defined as a linear convex combination of the subfaces and the interior vertex of the element, $\boldsymbol{x}_4$, see Figure B.16. After subdivision, two subelements have an edge defined by a NURBS and a numerical quadrature is designed using the parametrization (B.5). The third element has planar faces and a standard tetrahedral quadrature can be considered.

The second example considers an element Ω_e with two faces defined by different NURBS surfaces, as represented in Figure B.17. To perform the numerical integration in Ω_e the tetrahedral element is split in four *subelements* using its center of mass, $\boldsymbol{x}_C^E$. Each *subelement* is defined as a linear convex combination of $\boldsymbol{x}_C^E$ and an original face of Ω_e, having at most one face on the NURBS boundary. Therefore, numerical integration can be performed with the strategy proposed in Section 2.3 for curved

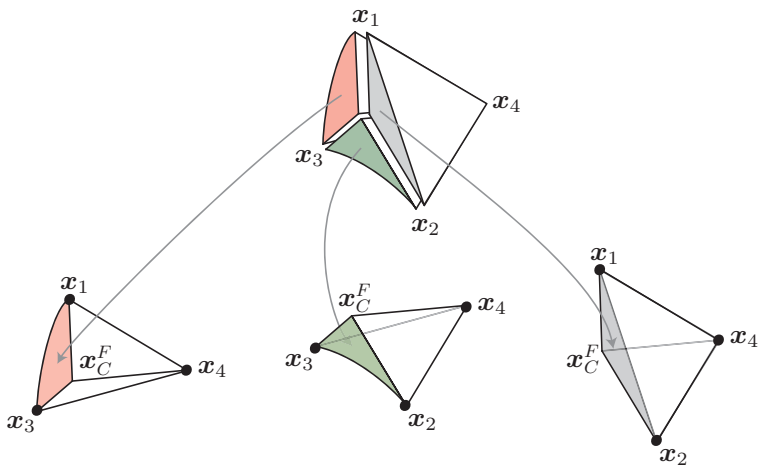


Figure B.16: Splitting of an element using the center of mass, $\boldsymbol{x}_C^F$, of a curved face with two edges defined by different NURBS

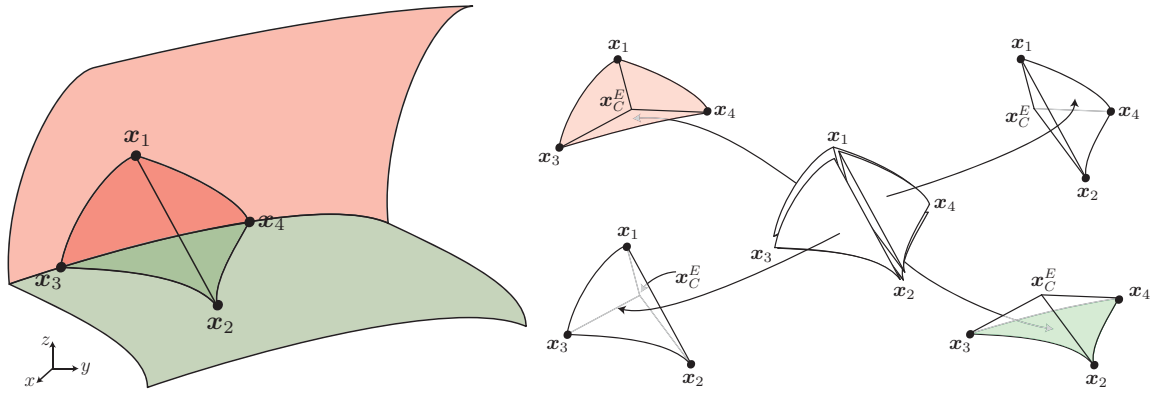


Figure B.17: Splitting of an element with two faces defined by different NURBS using its center of mass, $\boldsymbol{x}_C^E$

subelements, and by using standard tetrahedral quadratures for subelements with planar faces.

Finally, by combination of these two subdivision strategies, any element with several faces and/or edges on the NURBS boundary can be easily split into elements with only one face or one edge on the NURBS boundary. Thus, the design of a numerical quadrature is reduced to the cases presented in detail in Section 2.3 for an element with one face on the NURBS boundary, and in this section for an element with one edge defined by NURBS.

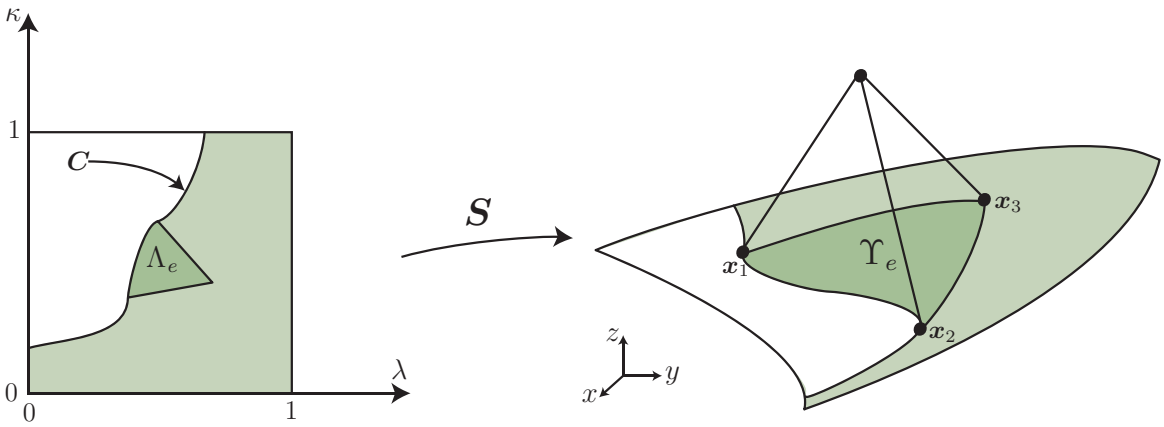


Figure B.18: Definition of a curved tetrahedral face on the NURBS boundary. The NURBS surface $\boldsymbol{S}$ is trimmed by the NURBS curve $\boldsymbol{C}$, leading to a curved triangle Λ_e in the parametric space

Trimmed and singular NURBS

The proposed strategy to design a quadrature on an element with a face on the NURBS boundary assumes a straight-sided parametric triangle Λ_e, see Section 2.3. This is true in most situations but special attention must be paid in more general situations involving trimmed or singular NURBS surfaces.

The parametric triangle Λ_e may be curved if trimmed NURBS surfaces are considered. In this case, the curved edges of Λ_e are NURBS curves (used to trim the original surface), see an example in Figure B.18. Therefore, an efficient strategy to design a numerical quadrature on Λ_e is to use the technique proposed by Sevilla and Fernández-Méndez (2009) for the numerical integration for NEFEM in 2D domains.

Finally, if the curved face contains a singular point of the NURBS parametrization, Λ_e must be defined as a quadrilateral in the parametric space of the NURBS, see an example in Figure B.19.

Obviously, the most complex situation involve the definition of a curved face defined by a trimmed singular NURBS, being necessary to consider Λ_e as a curved quadrilateral subdomain. In any case, the parametrization $\boldsymbol{\Psi}$ defined in Section 2.3 can be directly used to design a numerical quadrature on the curved element, without any modification. It is only necessary to change the definition of the numerical

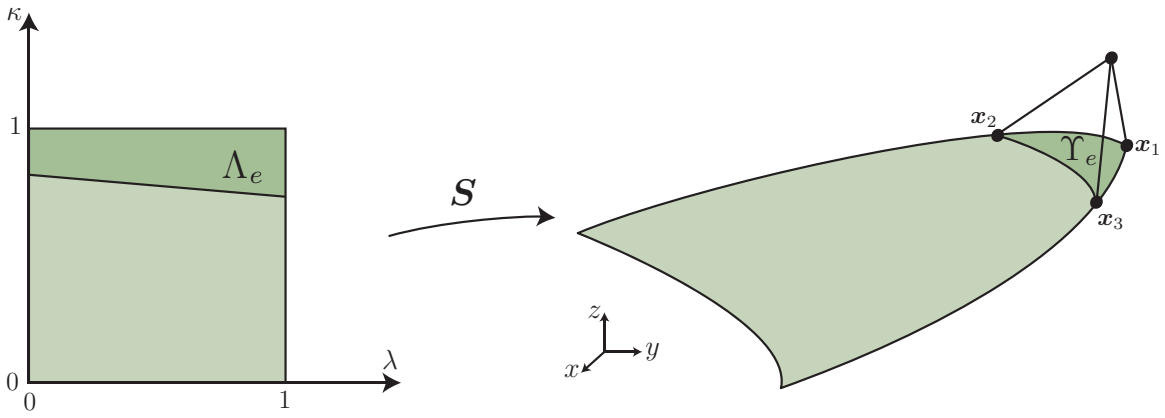


Figure B.19: Definition of a curved tetrahedral face on the NURBS boundary with a singular point, leading to a quadrilateral Λ_e in the parametric space

quadrature in Λ_e. The situation is more complex in the p-FEM, where a specific numerical quadrature must be designed in the reference tetrahedral, see more details in the comparison presented in Appendix C.

B.3 How to *NURBS-enhance* a finite element code?

The enhancement of an existing FE code with the NEFEM methodology requires little effort. Note that the main difference of a NEFEM code with a standard FE code is at the level of the computation of elemental matrices and vectors for curved elements and faces or edges. In fact, fortunately, the usual routines for the computation of elemental matrices and vectors for straight-sided elements can be directly used, without any modification. The usual inputs of these routines are the integration points and the shape functions evaluated at these points. In the case of curved elements intersecting the NURBS boundary, these inputs are computed as described in previous sections. Thus, most of the routines usual in a standard FE code (routines for assembly, computation of elemental matrices and vectors, etc) can be directly used.

The most crucial point in the implementation may be the inclusion of the NURBS boundary information. The information for the evaluation of all NURBS describing the boundary has to be stored. In 2D, for every curved edge the information of the

corresponding trimmed NURBS is necessary, that is, the extremes of the interval λ_1^e and λ_2^e and a pointer to the information of the NURBS curve $\boldsymbol{C}$. In 3D, for every curved face or edge, the parametric coordinates of each vertex are also necessary and a pointer to the information of the NURBS surface $\boldsymbol{S}$. Trimmed NURBS require an extra information in 3D domains, a pointer to the NURBS curve used to trim the original surface and the parametric coordinates of the vertices respect to the curve. Nowadays this is not an information usually provided by standard mesh generators but, it is worth noting that routines for the manipulation of NURBS can be easily obtained or implemented, see Piegl and Tiller (1995).

On the other hand, in the context of DG formulations, NEFEM is a natural option for the implementation of high-order approximations in domains with curved boundaries. In DG codes it is usual to store only the vertices of a triangle mesh, and their connectivities, usually obtained with a linear mesh generator. For high-order computations with non curved elements, if needed, all nodal coordinates are determined from the vertices coordinates. For isoparametric curved elements all the nodal coordinates of the element must be stored in order to define the mapping between local and cartesian coordinates. Under these circumstances NEFEM allows a straightforward implementation of curved boundaries, with no need of a high-order mesh generator, because the nodal coordinates at curved elements can be determined from the vertices of the element and the NURBS information. Moreover, the definition of the interpolation in cartesian coordinates implies that no specific nodal distributions are necessary to guarantee optimal convergence, see Section 2.4 and the numerical examples in this appendix.

B.4 Pre and postprocess

This section describes the pre and postprocess stages in a NEFEM computation. A strategy to obtain coarse meshes in domains with small geometric details is described. Finally, the visualization technique adopted in NEFEM for both an accurate postprocess of the geometry and the high-order solution is detailed.

B.4.1 Mesh generation

The full benefits of NEFEM require a mesh generation technology nowadays not available. The ability to compute accurate solutions in coarse meshes requires a mesh generator able to define coarse meshes of complex objects without h-refinement to capture small geometric features. In particular, current mesh generators do not allow to generate a mesh near a boundary corner without a node on the corner, see the NEFEM discretizations used in some examples in Chapter 3. This section is devoted to discuss a simple strategy to obtain coarse meshes suitable for NEFEM.

Given a CAD geometry containing small geometric details, the process to define a coarse NEFEM mesh starts with the definition of an *auxiliary CAD*. In this temporal CAD, geometric details are removed in order to avoid an excessive h-refinement, usually introduced by standard mesh generators. To illustrate the process, let us consider the example of Figure B.20. The exact CAD geometry is represented in Figure B.20 (a), corresponding to the *irregular* circle considered in Section 3.3. The auxiliary CAD geometry is represented in Figure B.20 (b), where the exact boundary is replaced by a polygonal shape. Note that, the polygonal shape is defined in terms of the desired element size. Then, a standard mesh generator is used to obtain a coarse mesh over the auxiliary CAD geometry, see Figure B.20 (c). Finally, the polygonal shape is replaced by the exact boundary representation, and the NEFEM mesh is obtained, see Figure B.20 (d).

Clearly, this simple strategy may lead to non-valid meshes when the polygonal shape is replaced by the exact boundary, see an example in Figure B.21. Therefore, it is necessary to develop efficient procedures to detect the validity of a NEFEM mesh and to correct non-valid meshes. Note that this problem is shared by several high-order FE methods in which coarse meshes are used and convergence is achieved by increasing p. For instance a detailed study of the mesh generation process for p-FEM is described by Luo et al. (2002). Recent advances in this area are presented by Persson and Peraire (2009). The authors propose a method for generating well-shaped curved unstructured meshes using a nonlinear elasticity analogy. In the mentioned works the solution is based on correction of a linear mesh by curving internal edges. Although in

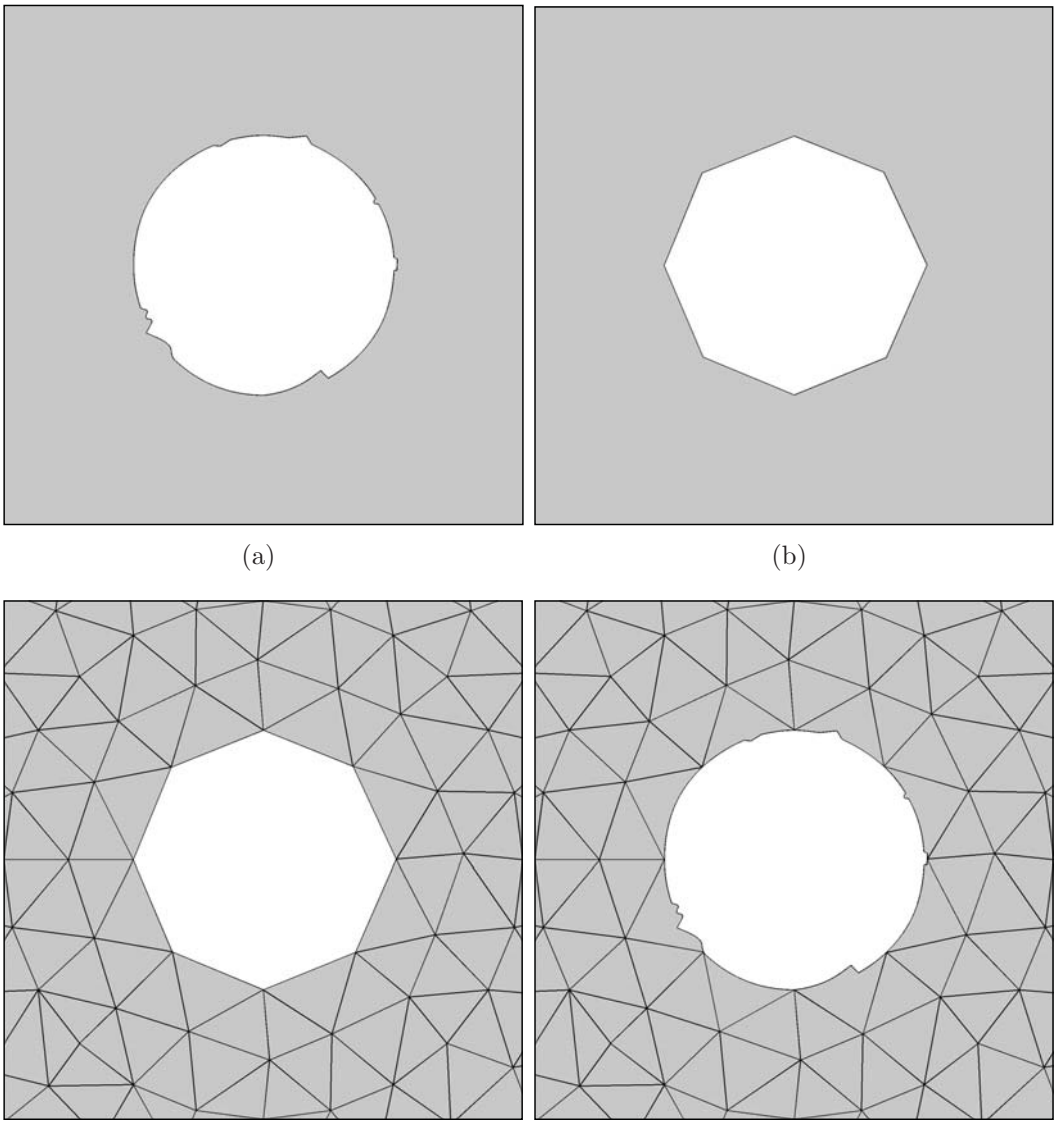


Figure B.20: Generation of a coarse mesh around an *irregular* circle without h-refinement to capture small geometric details: (a) exact CAD geometry, (b) auxiliary CAD geometry, (c) mesh over the auxiliary geometry, and (d) NEFEM mesh of the exact geometry

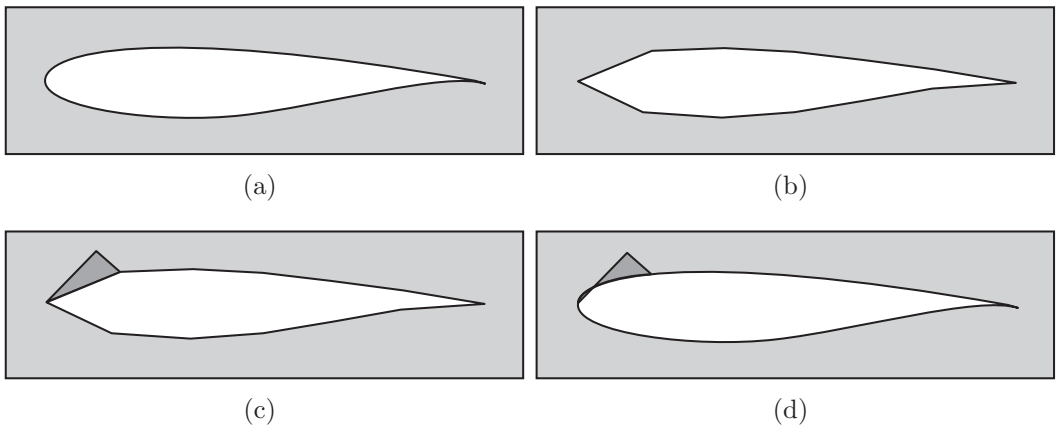


Figure B.21: Generation of a coarse NEFEM mesh around an airfoil leading to a non-valid element: (a) exact CAD geometry, (b) auxiliary CAD geometry, (c) mesh over the auxiliary geometry, and (d) NEFEM mesh of the exact geometry

CFD applications it is sometimes necessary to use curved internal edges to accurately capture the boundary layer. But it is important to note that internal curved edges are not mandatory to accurately capture the solution of electromagnetic scattering problems with high-order approximations. In fact, the use of straight interior edges lead to the minimum number of curved elements, and therefore, to a more efficient computation as it will be described in Section D.3.

B.4.2 Visualization technique

The use of high-order approximations usually means that coarse meshes are employed in which each element supports a high-order description of the numerical solution. As standard visualization packages only allow to represent linear surfaces it is necessary to develop specific tools for the representation of the curved NEFEM elements and also for the postprocess of the high-order solution.

To properly represent the exact geometry, the parametrizations used for numerical integration can be directly used for visualization purposes. For instance, consider a tetrahedral face on the NURBS boundary. The parametric triangle Λ_e is triangulated and the NURBS image of the triangulation points is used to define a surface mesh of

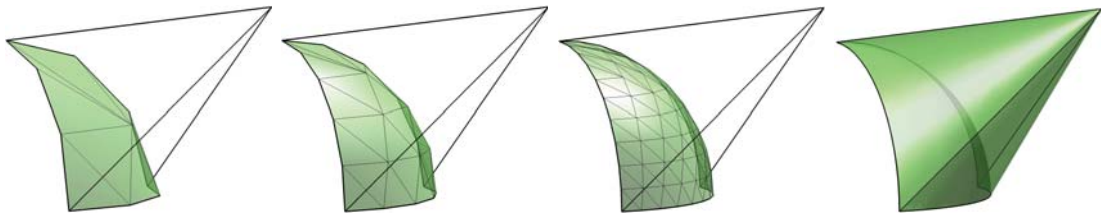


Figure B.22: Visualization of a curved tetrahedral face on the NURBS boundary for increasing postprocess resolution, and a curved NEFEM element

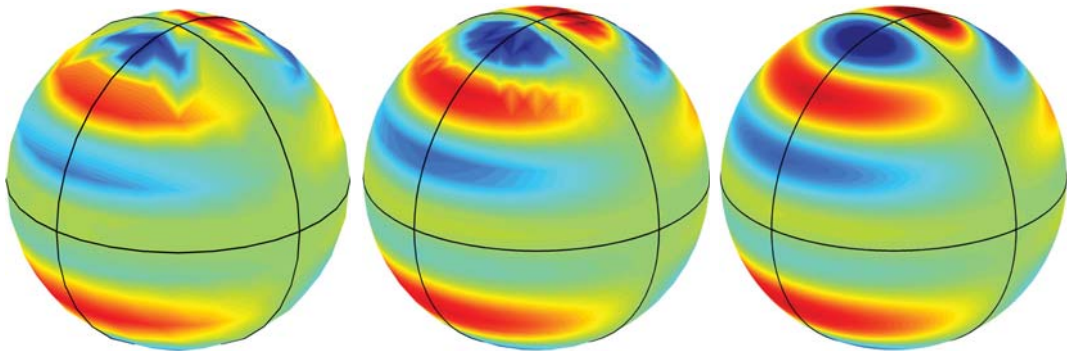


Figure B.23: Visualization of an electromagnetic field on curved tetrahedral faces defined by NURBS surfaces, for increasing postprocess resolution

the curved face, in the physical space. By refinement of the triangulation of Λ_e, the resolution of the final plot is increased, see Figure B.22. The same idea can be implemented for the visualization of curved faces with an edge on the NURBS boundary, or even for more complex situations. Figure B.22 also represents the internal faces of a curved tetrahedral element with a face on the NURBS boundary.

To provide an accurate representation of the high-order solution, it can be interpolated on the surface mesh used for an accurate description of the curved surfaces. For instance, Figure B.23 shows an electromagnetic field over some curved tetrahedral faces describing a sphere. The solution is computed with NEFEM and a degree of approximation $p = 8$, i.e. 45 nodes on each curved tetrahedral face. It is worth to mention that the first plot in Figure B.23 corresponds to the postprocess obtained using the 45 nodal values of the solution, offering a very poor description of the solution.

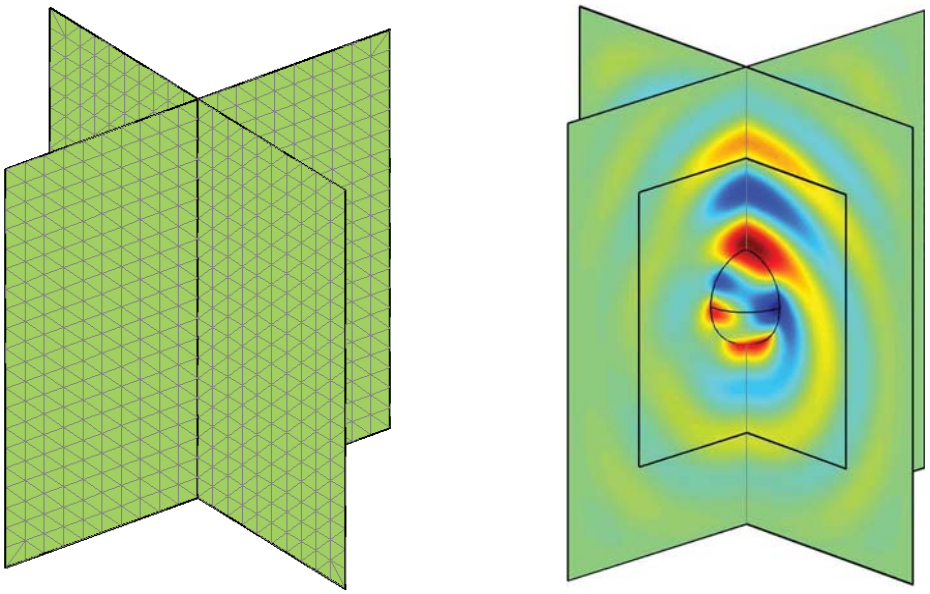


Figure B.24: Visualization of two sections of the numerical solution in a 3D NEFEM computation

As the triangulation used for the postprocess is refined, a higher quality representation of the high-order solution is obtained.

In three dimensional computations it is also useful to visualize the solution inside the domain, for instance some cuts of the 3D solution. For this purpose, a mesh is defined over the plane of interest, as represented in Figure B.24, and the solution is interpolated at nodes of the 2D mesh. For each node, the key issue is to determine its tetrahedral *father*, that is, the tetrahedral element containing this point, if exists. To reduce the search of a tetrahedral father, for a given point, a list of candidates can be easily defined in terms of the mesh size. Once the father is identified, the solution is interpolated. In NEFEM, the solution is interpolated directly in cartesian coordinates, whereas for isoparametric FEs or p-FEM the non-linear mapping relating local and cartesian coordinates must be inverted to interpolate the solution in local coordinates. Figure B.24 shows two meshes used to interpolate the solution, and the postprocess of the solution in these planes, showing an electromagnetic field inside the computational domain.

Note that the efficiency of the adopted visualization technique can be drastically improved by using adaptive refinement instead of uniform refinement, as proposed by Remacle et al. (2005).

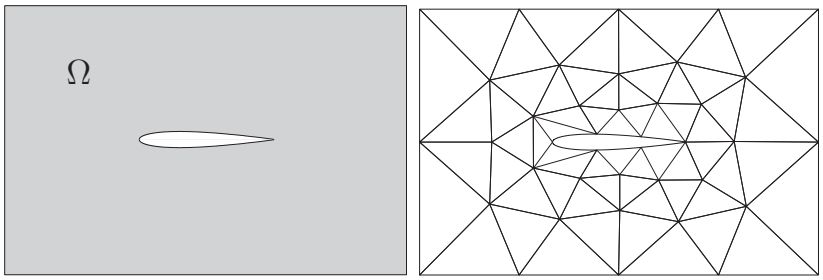
Appendix C

Comparison of high-order curved finite elements

An accurate geometrical description of a domain with curved boundaries is critical in the so-called *p extensions* of the FEM (Szabó and Babuška, 1991). In this approach the mesh remains unchanged (usually containing elements with a large aspect ratio) and the polynomial order of the approximation is increased in order to properly approximate the solution. In some applications, geometric errors introduced by the isoparametric mapping deteriorate the accuracy of the numerical solution, see Szabó and Babuška (1991), Xue and Demkowicz (2005) and citeIJNME-NEFEM. Therefore, an accurate description of the geometry is mandatory in order to obtain the maximum accuracy for a given spatial discretization. Thus, in p-FEM (Szabó et al., 2004), *blending functions* introduced by Gordon and Hall (1973) are usually considered to define an exact mapping relating local and cartesian coordinates.

This appendix is devoted to recall and compare several methodologies for the treatment of curved boundaries: isoparametric FEM and cartesian FEM, with an approximate description of the geometry, and p-FEM with an exact boundary representation. These methodologies have been used through the thesis to compare the performance of NEFEM for both standard Galerkin and DG formulations. To simplify the presentation, triangular elements with one curved side are considered.

Let $\Omega \subset \mathbb{R}^2$ be an open bounded domain whose boundary $\partial\Omega$, or a portion of it, is curved. A regular partition of the domain $\overline{\Omega} = \bigcup_e \overline{\Omega}_e$ in triangular elements



(a) (b)

Figure C.1: (a) Physical domain Ω with a curved boundary and (b) a triangulation of the domain with curved FE

is assumed, such that $\Omega_i \bigcap \Omega_j = \emptyset$, for $i \neq j$. For instance, Figure C.1 shows a domain with part of the boundary described by a NURBS curve corresponding to the NACA0012 airfoil, and a triangulation of the domain with curved FE. It is important to remark that, in the following, Ω_e denotes the element with an exact description of the curved boundary, also referred as *physical subdomain*. This is not the case of classical isoparametric FE, where the *computational element*, Ω_e^h, corresponds to a polynomial approximation of the curved boundary.

C.1 Isoparametric finite elements

The most widely used FE technique in the presence of curved boundaries is the isoparametric FEM, see Zienkiewicz (1971). A nodal interpolation of the solution, u, is considered in the reference element I with local coordinates $\boldsymbol{\xi} = (\xi, \eta)$, see Figure C.2,

$$u(\boldsymbol{\xi}) \simeq u^h(\boldsymbol{\xi}) = \sum_{i=1}^{\mathtt{n_{en}}} u_i N_i(\boldsymbol{\xi}), \tag{C.1}$$

where u_i are the nodal values, N_i are the polynomial shape functions of order p in $\boldsymbol{\xi}$, and $\mathtt{n_{en}}$ is the number of element nodes. The isoparametric transformation is used to

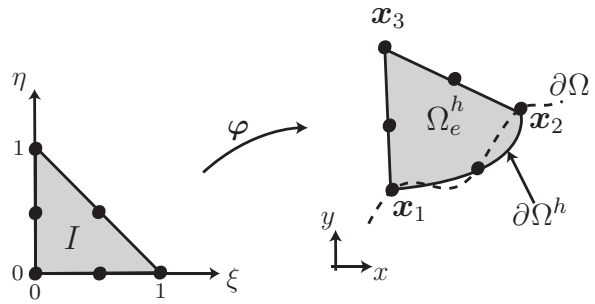


Figure C.2: Isoparametric mapping between the reference element I, in local coordinates $\boldsymbol{\xi}$, and an approximation of the physical subdomain in cartesian coordinates $\boldsymbol{x}$, namely $\Omega_e^h = \boldsymbol{\varphi}(I)$

relate local and cartesian coordinates

$$\begin{aligned} \boldsymbol{\varphi} \ &: I \longrightarrow \Omega_e^h \\ \boldsymbol{\xi} &\longmapsto \boldsymbol{\varphi}(\boldsymbol{\xi}) := \sum_{i=1}^{\mathtt{n_{en}}} \boldsymbol{x}_i N_i(\boldsymbol{\xi}), \end{aligned} \tag{C.2}$$

where $\boldsymbol{x}_i$ are the nodal coordinates of the computational element Ω_e^h. Note that Ω_e^h is the approximation of the physical subdomain Ω_e with a polynomial approximation of the boundary, see Figure C.2. In fact, the term *isoparametric* stands for the use of the same polynomial shape functions to define the functional approximation, and to describe the geometry of the computational element in cartesian coordinates, see Equation (C.2).

Numerical integration in the computational element Ω_e^h (approximation of Ω_e) is performed using the isoparametric transformation (C.2), with a numerical quadrature in I. For instance, a stiffness elemental matrix coefficient is computed as

$$K_{ij}^e = \int_{\Omega_e^h} \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{\xi}(\boldsymbol{x})\big) \cdot \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{\xi}(\boldsymbol{x})\big)\, d\Omega = \int_I \Big(\boldsymbol{J}_{\boldsymbol{\varphi}}^{-1} \boldsymbol{\nabla}_{\boldsymbol{\xi}} N_i(\boldsymbol{\xi})\Big) \cdot \Big(\boldsymbol{J}_{\boldsymbol{\varphi}}^{-1} \boldsymbol{\nabla}_{\boldsymbol{\xi}} N_j(\boldsymbol{\xi})\Big) |\boldsymbol{J}_{\boldsymbol{\varphi}}|\, d\boldsymbol{\xi}, \tag{C.3}$$

where $\boldsymbol{J}_{\boldsymbol{\varphi}}$ is the Jacobian of the isoparametric transformation. For curved elements the isoparametric mapping is non-linear. Therefore, the inverse of the Jacobian, $\boldsymbol{J}_{\boldsymbol{\varphi}}^{-1}$, is not a polynomial function, and no exact integration is feasible with standard quadrature rules. In practice, a symmetric triangle quadrature Wandzura and Xiao (2003) on I, with a sufficiently large number of integration points, is usually employed to compute integral (C.3). In fact, a quadrature of order $2p-1$ provides optimal convergence of the isoparametric FEM, see Zienkiewicz and L.Taylor (2000).

There are two focus of error in the isoparametric FEM. First, the isoparametric mapping (C.2) introduces geometric errors, due to the approximation of the physical subdomain Ω_e by the computational element Ω_e^h. In fact, the boundary of the computational domain $\partial\Omega_h$ is a piecewise polynomial approximation of the exact boundary $\partial\Omega$, see Figure C.2. On the other hand, for high-order approximations on curved elements, the definition of the polynomial interpolation (C.1) in local coordinates, $\boldsymbol{\xi}$, implies a loss of consistency: a polynomial interpolation of degree $p>1$ in $\boldsymbol{\xi}$ does not correspond to a polynomial interpolation of degree p in $\boldsymbol{x}$. This means that the approximation is able to reproduce linear functions but it is not able to reproduce higher order polynomials in cartesian coordinates. In other words, curved isoparametric FE pass the patch test but they fail to pass the so-called *higher order patch tests*, see Zienkiewicz and L.Taylor (2000) for further details.

Remark 1. *Optimal convergence of isoparametric FE is obtained under some smoothness assumptions on the isoparametric mapping. In practice, a specific node placement of interior nodes in curved elements of order $p>2$ is mandatory to guarantee optimal rates of convergence, see Ciarlet and Raviart (1972) and Lenoir (1986).*

C.2 *Cartesian* finite elements

An alternative to ensure consistency of the approximation, and optimal convergence for any nodal distribution, is the so-called cartesian FEM. In this approach, the polynomial basis for the approximation of the solution is defined with cartesian coordinates

$\boldsymbol{x}$

$$u(\boldsymbol{x}) \simeq u^h(\boldsymbol{x}) = \sum_{i=1}^{\mathtt{n}_{\mathtt{en}}} u_i N_i(\boldsymbol{x}).$$

Nevertheless, the isoparametric transformation (C.2) and the computational element Ω_e^h, are still considered for integration purposes. For instance, a stiffness elemental matrix coefficient is computed as

$$K_{ij}^e = \int_{\Omega_e^h} \boldsymbol{\nabla}_{\boldsymbol{x}} N_i(\boldsymbol{x}) \cdot \boldsymbol{\nabla}_{\boldsymbol{x}} N_j(\boldsymbol{x})\, d\Omega = \int_I \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{x}(\boldsymbol{\xi})\big) \cdot \boldsymbol{\nabla}_{\boldsymbol{x}} N_j\big(\boldsymbol{x}(\boldsymbol{\xi})\big) |\boldsymbol{J}_{\boldsymbol{\varphi}}|\, d\boldsymbol{\xi}. \qquad \text{(C.4)}$$

The definition of the polynomial basis for the approximation with cartesian coordinates, $\boldsymbol{x}$, ensures reproducibility of polynomials, i.e. consistency of order p. Moreover, exact integration is feasible because shape functions are polynomials, not only in cartesian coordinates $\boldsymbol{x}$, but also on local coordinates $\boldsymbol{\xi}$. More precisely, for a degree of interpolation p, $N_i\big(\boldsymbol{x}(\boldsymbol{\xi})\big)$ is a polynomial of degree p^2, and the function to be integrated in local coordinates, $f(\boldsymbol{\xi}) = \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{x}(\boldsymbol{\xi})\big) \cdot \boldsymbol{\nabla}_{\boldsymbol{x}} N_j\big(\boldsymbol{x}(\boldsymbol{\xi})\big) |\boldsymbol{J}_{\boldsymbol{\varphi}}|$, is a polynomial of degree $2p(p+1)$. Therefore, integral (C.4) can be exactly computed with a triangle quadrature of order $2p(p+1)$ on the reference element I. Nevertheless, the integration is still done in the (approximated) computational element Ω_e^h. Thus, although cartesian FEM gets rid of the consistency lack of isoparametric FEM, it still maintains the geometric error. This is not the case for p-FEM, with the exact boundary representation, described in next section.

C.3 p-version finite elements

In p-FEM, the nodal interpolation is defined in the reference element I with local coordinates $\boldsymbol{\xi}$, see Equation (C.1), but an exact mapping between the reference element I and the physical subdomain Ω_e is employed. For instance, assuming a NURBS

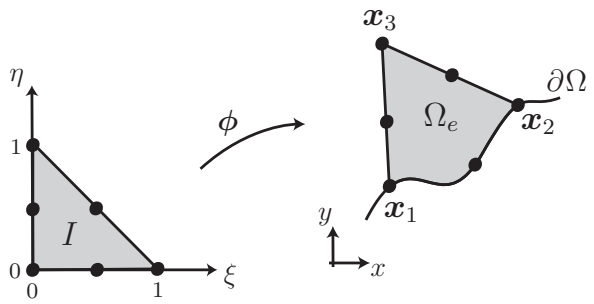


Figure C.3: Exact mapping between the reference element I with local coordinates $\boldsymbol{\xi}$, and the physical subdomain Ω_e with cartesian coordinates $\boldsymbol{x}$

parametrization $\boldsymbol{C}(\xi)$ of the curved edge of Ω_e, a p-FEM mapping is

$$\begin{aligned} \boldsymbol{\phi} \ &: I \longrightarrow \Omega_e \\ \boldsymbol{\xi} &\longmapsto \boldsymbol{\phi}(\boldsymbol{\xi}) := \frac{1-\xi-\eta}{1-\xi}\boldsymbol{C}(\xi) + \frac{\xi\eta}{1-\xi}\boldsymbol{x}_2 + \eta\boldsymbol{x}_3, \end{aligned} \tag{C.5}$$

where $\boldsymbol{x}_1 = \boldsymbol{C}(0)$ and $\boldsymbol{x}_2 = \boldsymbol{C}(1)$ are the vertices of Ω_e on the curved boundary, and $\boldsymbol{x}_3$ is the internal vertex, see Figure C.3. Other options to define an exact mapping from I to Ω_e are possible, see for instance Pascal and George (1999). However, no relevant differences are observed in the numerical examples presented in this thesis.

In p-FEM, a stiffness elemental matrix coefficient is computed as

$$K_{ij}^e = \int_{\Omega_e} \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{\xi}(\boldsymbol{x})\big) \cdot \boldsymbol{\nabla}_{\boldsymbol{x}} N_i\big(\boldsymbol{\xi}(\boldsymbol{x})\big)\, d\Omega = \int_I \Big(\boldsymbol{J}_{\boldsymbol{\phi}}^{-1}\boldsymbol{\nabla}_{\boldsymbol{\xi}} N_i(\boldsymbol{\xi})\Big) \cdot \Big(\boldsymbol{J}_{\boldsymbol{\phi}}^{-1}\boldsymbol{\nabla}_{\boldsymbol{\xi}} N_j(\boldsymbol{\xi})\Big) |\boldsymbol{J}_{\boldsymbol{\phi}}|\, d\boldsymbol{\xi},$$

integrating over the physical subdomain Ω_e, with an exact description of the geometry. Note that, the inverse of the Jacobian, $\boldsymbol{J}_{\boldsymbol{\phi}}^{-1}$, is not a polynomial function and, as for the isoparametric FEM, no exact integration is feasible with standard quadrature rules. Nevertheless, under some smoothness requirements on the parametrization $\boldsymbol{C}(\xi)$, the same quadrature order used in the isoparametric FEM, that is $2p-1$, guarantees

	Exact geometry	Consistency
Isoparametric FEM	NO	NO
Cartesian FEM	NO	YES
p-FEM	YES	NO
NEFEM	YES	YES

Table C.1: Comparison of FE techniques used in domains with curved boundaries

optimal convergence, see Banerjee and Suri (1992).

Note that p-FEM presents the major advantage, in front of isoparametric or cartesian FE, of an exact boundary representation. Nevertheless, p-FEM still suffers the same lack of consistency as isoparametric FEM, due to the definition of the polynomial shape functions in the reference element I, with local coordinates $\boldsymbol{\xi}$. This is not the case for NEFEM.

C.4 Comparison

This section presents a critical comparison of several techniques to treat curved boundaries. NEFEM is compared with isoparametric FEM, cartesian FEM and p-FEM. The main differences between these FE techniques are summarized in Table C.1. The use of a non-linear mapping relating local and cartesian coordinates (isoparametric transformation in the isoparametric FEM and an exact mapping in the p-FEM) induces a loss of consistency: a polynomial interpolation of degree $p > 1$ in local coordinates $\boldsymbol{\xi}$, does not correspond to a polynomial interpolation of degree p in cartesian coordinates $\boldsymbol{x}$. On the other hand, the use of the isoparametric mapping to perform the numerical integration (as done in the isoparametric FEM and in the cartesian FEM) introduces geometric errors: the boundary of the computational domain, $\partial\Omega_h$, is a piecewise polynomial approximation of the exact boundary, $\partial\Omega$. The only method ensuring consistency of the approximation (for any p) and an exact boundary representation of the domain is NEFEM, see Table C.1.

It is worth mentioning that, from a computational point of view, the definition of the polynomial basis in local coordinates $\boldsymbol{\xi}$, as done in the isoparametric FEM and

in the p-FEM, induces a marginal extra efficiency. In this case the polynomial basis is defined once in the reference element and used to define the approximation in each curved element, whereas a cartesian approximation requires a specific definition of the polynomial basis for each curved element. Nevertheless, it is important to recall that the extra cost associated to the basis definition in cartesian coordinates $\boldsymbol{x}$, is restricted to elements intersecting the NURBS boundary, in real applications a very small portion of the total number of elements.

A priori error estimates for the FE methodologies considered in this work have similar expressions, with optimal convergence in all cases. However, the hypothesis to obtain these estimates are different, depending on the definition of the approximation, in local or cartesian coordinates, and on the boundary representation, that is approximated or exact.

When the polynomial basis is defined with local coordinates $\boldsymbol{\xi}$, the mapping relating local and cartesian coordinates must be smooth enough to guarantee optimal convergence. In practice, for the isoparametric FEM specific nodal distributions on curved elements are necessary to obtain optimal convergence rates with $p > 2$, see **?** and **?**Lenoir (1986). For p-FEM the NURBS parametrization of the curved boundary must be smooth enough to guarantee the necessary smoothness of the p-FEM mapping relating local and cartesian coordinates, see Babuška and Suri (1987). In contrast, when the polynomial basis is defined with cartesian coordinates $\boldsymbol{x}$, the derivation of *a priori* error estimates is very close to FE *a priori* error estimates in polygonal domains, which can be found in Johnson (1987) and Brenner and Scott (1994). For cartesian FEM and NEFEM no specific nodal distributions in curved elements are necessary to achieve optimal convergence. Moreover, smooth variations of the NURBS parametrization are not required to obtain the optimal convergence rates in NEFEM.

Nevertheless, optimal *a priori* error estimates for FE methods with a cartesian approximation of the solution requires an extra hypothesis if a strong imposition of Dirichlet boundary conditions is considered. In such situation optimal nodal distributions on curved boundaries are necessary for an accurate interpolation of Dirichlet

boundary conditions, see Section 2.4. Moreover, when internal curved edges or faces are present in the mesh, optimal nodal sets are needed to keep the optimal convergence rates, see Section 2.4 and the discussion in Appendix B.1.

Note that, for FE methods with an approximate boundary representation (isoparametric FEM and cartesian FEM) optimal convergence is provided under the assumption that geometric errors are lower than the discretization error, that is, the difference between the computational element Ω_e^h and the physical subdomain Ω_e is bounded by γh^p, where γ is a constant, h is the mesh size and p is the interpolation degree. Moreover, bounds of the Jacobian of the isoparametric transformation and its first p derivatives are also necessary (Ciarlet and Raviart, 1972). Thus, a curved element with an approximated boundary representation must verify two contradictory requirements. On one hand, the computational polynomial boundary has to be close enough to the curved boundary. And on the other hand, the discrepancy between the curved element and the straight element given by its vertices must vanish fast enough (Lenoir, 1986).

Finally, it is worth to recall that NURBS are piecewise rational functions defined in parametric form (Piegl and Tiller, 1995). Therefore, numerical integration for p-FEM and NEFEM should be designed to account for changes of NURBS definition along the curved edge of Ω_e, see the next section.

C.4.1 Numerical integration for p-FEM and NEFEM

This section discusses the numerical integration for p-FEM and NEFEM when changes of NURBS parametrization are considered inside the boundary curved edge of an element Ω_e in 2D, or a boundary face in 3D.

For illustration purposes the triangle with a curved edge represented in Figure C.4 is considered first. The curved edge is described with a piecewise rational parametrization $\boldsymbol{C}$, whose definition changes in two points on the curved edge, marked with $\square$. The parametric coordinates of these points are called the *breakpoints* or *knots* of the NURBS parametrization, see Piegl and Tiller (1995).

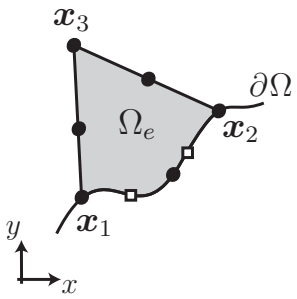


Figure C.4: Triangle with a curved edge containing changes of NURBS definition (marked with □)

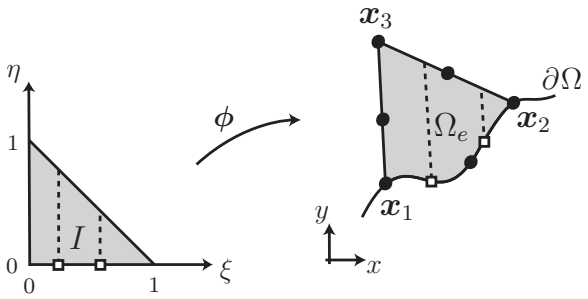


Figure C.5: Numerical integration for 2D p-FEM: subdivision of the reference element I to design a numerical quadrature taking into account changes of NURBS parametrization $\boldsymbol{C}(\xi)$ at points marked with □

In p-FEM, the piecewise definition of the boundary induces a piecewise definition of the mapping $\boldsymbol{\phi}$, see Equation (C.5). Therefore, a specifically designed numerical quadrature must be considered in the reference element I. For the triangle represented in Figure C.4, with two changes of NURBS definition, the reference element should be partitioned as represented in Figure C.5, where the discontinuous lines show the changes of definition of the mapping $\boldsymbol{\phi}$. Note that these lines origin at the breakpoints of the NURBS parametrization in the ξ axis, and are extended inside the reference element. A composite numerical quadrature on I should be defined by using different numerical quadratures in each region. An efficient option is to consider a triangle quadrature and a tensor product of 1D Gaussian quadratures in quadrilateral regions.

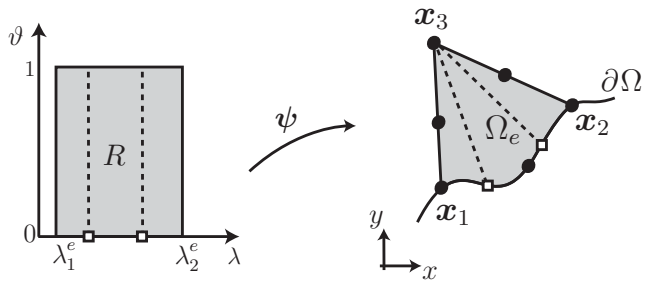
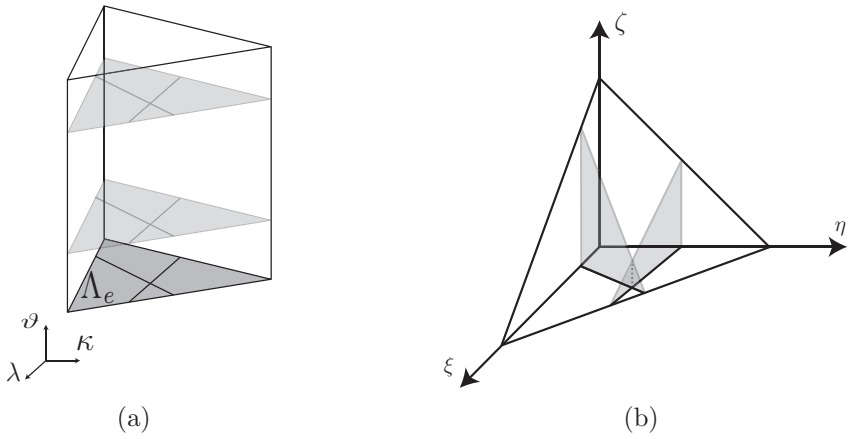


Figure C.6: Numerical integration for 2D NEFEM: subdivision of the rectangle $R = [\lambda_1^e, \lambda_2^e] \times [0, 1]$ to design a numerical quadrature taking into account changes of NURBS parametrization $\boldsymbol{C}(\lambda)$ at points marked with $\square$

In NEFEM, changes of NURBS definition are easily accommodated using application $\boldsymbol{\psi}$, see Equation (B.3). The piecewise definition of the boundary also induces a piecewise definition of the mapping $\boldsymbol{\psi}$. The rectangle R is subdivided using the breakpoints, as represented in Figure C.6, and a numerical quadrature in R is defined in terms of 1D Gaussian quadratures. A composite 1D Gauss quadrature is used in parameter λ to take into account the discontinuous nature of the NURBS parametrization. In the other parameter, ϑ, exact integration is feasible as commented in Section B.2.1.

For NEFEM, the proposed strategy to perform the numerical integration in 2Ds can be easily extended to 3D, see the details in Sections 2.3 and B.2. Changes of NURBS parametrization inside a curved face are easily treated in NEFEM. The parametric triangle Λ_e is subdivided according to the changes of NURBS parametrization, and numerical quadratures are defined in each subregion, see Figure C.7 (a) and Section 2.3.

For p-FEM in 3D, the definition of a numerical quadrature on the reference tetrahedral accounting changes of NURBS definition is more complicated. The generalization of the strategy adopted in 2D requires the subdivision of the reference tetrahedral element to account changes of NURBS surface parametrization, see an example in Figure C.7 (b). In this example a tetrahedron, a hexahedron and two prisms are obtained after subdivision, but, in general, other subregions are possible. Thus, a simple option



(a) (b)

Figure C.7: Subdivisions to design a numerical quadrature taking into account changes of NURBS definition: (a) on the prism $\Lambda_e \times [0,1]$ for 3D NEFEM, and (b) on the reference tetrahedral for 3D p-FEM

to define a quadrature on the reference element is to use further subdivision to obtain only tetrahedral subregions. Then, a composite quadrature may be defined on the reference element based on standard tetrahedral quadratures. In fact, a usual practice to facilitate the implementation of p-FEM in 3D is to consider a polynomial approximation of the boundary. For instance, in Coyle and Ledger (2005) a least-squares approximation of the exact boundary is considered in a p-FEM context. Although the polynomial approximation of the boundary can be selected to satisfy continuity requirements across element interfaces, see Luo et al. (2001), the exact boundary representation is no longer maintained for an ease of implementation.

Appendix D

Discontinous Galerkin methods for electromagnetic scattering

Computational electromagnetics cover a wide range of applications in different areas, like aeronautics, medicine, ground floor detection and optics. In the context of electromagnetic scattering, nowadays many applications need to simulate high frequency problems, in which classical methods require extremely refined meshes to overcome the dispersion associated with the propagation of the waves, see Morgan et al. (2000) or Hachemi et al. (2004). Reference Ainsworth (2004) quantifies the dispersion that occurs in high frequency wave propagation, also showing that it is best overcome by the use of high order methods. Consequently, many authors have focused their interest in the numerical solution of Maxwell's equations by means of high-order elements, see, among others, Ganesh and Graham (2004), Chen et al. (2005) and Ledger and Morgan (2005). In particular, DG methods, see a recent review by Cockburn (2004), have become very popular in the last years for the solution of Maxwell's equations in the time-domain, see Hesthaven and Warburton (2002), Kabakian et al. (2004) and Chen et al. (2005) among others.

In this appendix the application of DG methods to electromagnetic scattering problems is reviewed. First, the transient Maxwell's equations and the electromagnetic scattering problem are recalled. Boundary conditions arising in electromagnetic scattering applications are detailed. Secondly, the DG formulation of the conservative form of Maxwell's equations is reviewed. Particular emphasis is placed on the weakly

imposition of boundary conditions.

D.1 Electromagnetic scattering problems

For a linear isotropic material of relative permittivity ε and relative permeability μ, and assuming that there are no current sources in the material, the time dependent Maxwell's equations in differential dimensionless form become

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\boldsymbol{\nabla} \times \boldsymbol{E}^t \tag{D.1}$$

$$\frac{\partial \boldsymbol{D}}{\partial t} = \boldsymbol{\nabla} \times \boldsymbol{H}^t \tag{D.2}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{D} = 0 \tag{D.3}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{B} = 0, \tag{D.4}$$

where $\boldsymbol{E}^t$ is the total electric field, $\boldsymbol{H}^t$ the total magnetic field, $\boldsymbol{D}$ the electric flux density and $\boldsymbol{B}$ the magnetic flux density. This set of equations is closed with the constitutive laws

$$\boldsymbol{D} = \varepsilon \boldsymbol{E}^t, \qquad \boldsymbol{B} = \mu \boldsymbol{H}^t.$$

Remark 2. *Divergence conditions,* (D.3) *and* (D.4)*, are redundant because they can be derived from the curl equations* (D.1) *and* (D.2)*, provided that initial conditions are divergence-free.*

Electromagnetic scattering problems model the interaction between an incident electromagnetic wave and a general scatterer in an unbounded domain Ω^∞. The total electric and magnetic fields, $\boldsymbol{E}^t$ and $\boldsymbol{H}^t$ respectively, are usually split into incident and scattered field components

$$\boldsymbol{E}^t = \boldsymbol{E}^i + \boldsymbol{E}, \qquad \boldsymbol{H}^t = \boldsymbol{H}^i + \boldsymbol{H},$$

where superscript i refers to the incident field, and $\boldsymbol{E} = (E_1, E_2, E_3)^T$ and $\boldsymbol{H} =$

$(H_1, H_2, H_3)^T$ are the *scattered* electric and magnetic field intensity vectors. The incident field is chosen to satisfy Maxwell's equations in free space and, therefore, the numerical scheme is designed to solve only the scattered fields. This is usually referred to as the *scattered field formulation* (Taflove, 1995). Note that a major advantage of the scattered field formulation is that the incident field is imposed exactly on the surface of the scatterer. Whereas a total field formulation needs to propagate the incident field through the mesh to reach the obstacle, and the imposed field on the scatterer suffers from dispersion errors.

Maxwell's curl equations for the scattered field can be written as a linear hyperbolic system of conservation laws

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_k(\boldsymbol{U})}{\partial x_k} = \boldsymbol{S}(\boldsymbol{U}), \tag{D.5}$$

where Einstein summation notation is assumed. In 3D the vector of conserved quantities $\boldsymbol{U}$ and the fluxes $\boldsymbol{F}_k$ are

$$\boldsymbol{U} = \begin{pmatrix} \varepsilon \boldsymbol{E} \\ \mu \boldsymbol{H}^t \end{pmatrix} = \begin{pmatrix} \varepsilon E_1 \\ \varepsilon E_2 \\ \varepsilon E_3 \\ \mu H_1 \\ \mu H_2 \\ \mu H_3 \end{pmatrix}, \; \boldsymbol{F}_1 = \begin{pmatrix} 0 \\ H_3 \\ -H_2 \\ 0 \\ -E_3 \\ E_2 \end{pmatrix}, \; \boldsymbol{F}_2 = \begin{pmatrix} -H_3 \\ 0 \\ H_1 \\ E_3 \\ 0 \\ -E_1 \end{pmatrix}, \; \boldsymbol{F}_3 = \begin{pmatrix} H_2 \\ -H_1 \\ 0 \\ -E_2 \\ E_1 \\ 0 \end{pmatrix},$$

and the source term $\boldsymbol{S}$ is given by

$$\boldsymbol{S} = \begin{pmatrix} (1-\varepsilon)\dfrac{\partial \boldsymbol{E}^i}{\partial t} \\ (1-\mu)\dfrac{\partial \boldsymbol{H}^i}{\partial t} \end{pmatrix}.$$

In 2D, the hyperbolic system (D.5) decouples into the *Transverse Electric* (TE) and *Transverse Magnetic* (TM) modes. The vector of conserved quantities, the fluxes

and the source are given by

$$\boldsymbol{U} = \begin{pmatrix} \varepsilon E_1 \\ \varepsilon E_2 \\ \mu H_3 \end{pmatrix}, \ \boldsymbol{F}_1 = \begin{pmatrix} 0 \\ H_3 \\ E_2 \end{pmatrix}, \ \boldsymbol{F}_2 = \begin{pmatrix} -H_3 \\ 0 \\ -E_1 \end{pmatrix}, \ \boldsymbol{S} = \begin{pmatrix} (1-\varepsilon)\dfrac{\partial E_1^i}{\partial t} \\ (1-\varepsilon)\dfrac{\partial E_2^i}{\partial t} \\ (1-\mu)\dfrac{\partial H_3^i}{\partial t} \end{pmatrix},$$

for the TE mode, and

$$\boldsymbol{U} = \begin{pmatrix} \mu H_1 \\ \mu H_2 \\ \varepsilon E_3 \end{pmatrix}, \ \boldsymbol{F}_1 = \begin{pmatrix} 0 \\ -E_3 \\ -H_2 \end{pmatrix}, \ \boldsymbol{F}_2 = \begin{pmatrix} E_3 \\ 0 \\ H_1 \end{pmatrix}, \ \boldsymbol{S} = \begin{pmatrix} (1-\mu)\dfrac{\partial H_1^i}{\partial t} \\ (1-\mu)\dfrac{\partial H_2^i}{\partial t} \\ (1-\varepsilon)\dfrac{\partial E_3^i}{\partial t} \end{pmatrix}, \tag{D.6}$$

for the TM mode. Note that in free-space the source term vanishes because $\varepsilon = \mu = 1$.

For plane wave scattering the electric incident field is expressed as

$$\boldsymbol{E}^i = \boldsymbol{p}\cos(wt - \boldsymbol{k}\cdot\boldsymbol{x})$$

where $\boldsymbol{x} = (x, y, z)$ is the position vector, $\omega = 2\pi/\lambda$ is the angular frequency, and the vectors $\boldsymbol{p}$ and $\boldsymbol{k}$ are defined as

$$\boldsymbol{k} = \omega \begin{pmatrix} \cos\phi^i \sin\theta^i \\ \sin\phi^i \sin\theta^i \\ \cos\theta^i \end{pmatrix}, \quad \boldsymbol{p} = \begin{pmatrix} E_\theta^i \cos\theta^i \cos\phi^i - E_\phi^i \sin\phi^i \\ E_\theta^i \cos\theta^i \sin\phi^i + E_\phi^i \cos\phi^i \\ -E_\theta^i \sin\theta^i \end{pmatrix}.$$

The selection of the parameters E_ϕ^i, E_θ^i and the angles ϕ^i and θ^i (usually in spherical coordinates) determine the polarization of the incident wave, see Ledger et al. (2007) for more details.

Remark 3. *Although divergence conditions* (D.3) *and* (D.4) *are redundant at a con-*

tinuous level, see Remark 2, they can not be neglected in a numerical method. If they are neglected, the numerical solution of (D.5) results in computed fields that exhibit an offset caused by a spurious component. In this work, the offset is avoided by using a modified incident field, defined by

$$\boldsymbol{E}^i = \begin{cases} \mathbf{0} & wt \leq \boldsymbol{k}\cdot\boldsymbol{x} \\ \dfrac{wt - \boldsymbol{k}\cdot\boldsymbol{x}}{2\pi}\boldsymbol{p}\cos(wt - \boldsymbol{k}\cdot\boldsymbol{x}) & 0 < wt - \boldsymbol{k}\cdot\boldsymbol{x} \leq 2\pi \\ \boldsymbol{p}\cos(wt - \boldsymbol{k}\cdot\boldsymbol{x}) & 2\pi < wt - \boldsymbol{k}\cdot\boldsymbol{x}. \end{cases}$$

This adjustment is designed to make the spurious field vanish when the incident field reaches the correct value at the complete surface of the scatterer, see further details in Kangro and Nicolaides (1997). This approach is used in the numerical examples shown in Chapter 3.

D.1.1 Boundary conditions and interfaces

Two boundary conditions are usual in electromagnetic scattering applications: boundary conditions at material interfaces, such as the interface between free-space and the scatterer, and boundary conditions at infinity.

Let us consider an interface between two materials. The boundary condition across such an interfaces is

$$\boldsymbol{n}\times\boldsymbol{E}^L = \boldsymbol{n}\times\boldsymbol{E}^R, \tag{D.7}$$

$$\boldsymbol{n}\times\boldsymbol{H}^L = \boldsymbol{n}\times\boldsymbol{H}^R, \tag{D.8}$$

$$\boldsymbol{n}\cdot(\varepsilon^L\boldsymbol{E}^L) = -\boldsymbol{n}\cdot(\varepsilon^R\boldsymbol{E}^R) \tag{D.9}$$

$$\boldsymbol{n}\cdot(\mu^L\boldsymbol{H}^L) = -\boldsymbol{n}\cdot(\varepsilon^R\boldsymbol{H}^R) \tag{D.10}$$

Conditions (D.7) and (D.8) are appropriate when considering the Maxwell's curl equa-

tions (D.1) and (D.2). Equations (D.9) and (D.10) are the corresponding condition for the Maxwell's divergence equations (D.3) and (D.4). Since the conservation form (D.5) is just an alternative form for expressing the Maxwell's curl equations, only the conditions (D.7) and (D.8) must be considered, expressing the continuity of the tangential electric field across the interface. The superscripts L and R indicate the field values on either side of the interface, and $\boldsymbol{n}$ denotes the outward unit normal in the direction from L to R.

In several applications, the scatterer consists of a highly (electrically) conducting material that prevents the penetration of the electromagnetic fields inside. Such scatterers are called perfect electric conductors (PECs). On the surface of a PEC scatterer, the boundary conditions are

$$\begin{aligned} \boldsymbol{n} \times \boldsymbol{E} &= -\boldsymbol{n} \times \boldsymbol{E}^i, \\ \boldsymbol{n} \cdot \boldsymbol{H} &= -\boldsymbol{n} \cdot \boldsymbol{H}^i \end{aligned} \tag{D.11}$$

Again, only the condition (D.11) is appropriate when considering the Maxwell's curl equations. The implementation of boundary conditions at material interfaces is detailed in Section D.2.1.

To ensure uniqueness of the solution in an exterior domain, electric and magnetic fields must satisfy the Silver-Müller radiation conditions

$$\lim_{r\to\infty} \left(\boldsymbol{x} \times (\nabla \times \boldsymbol{E}) + r \frac{\partial \boldsymbol{E}}{\partial t} \right) = \boldsymbol{0}, \qquad \lim_{r\to\infty} \left(\boldsymbol{x} \times (\nabla \times \boldsymbol{H}) + r \frac{\partial \boldsymbol{H}}{\partial t} \right) = \boldsymbol{0}, \tag{D.12}$$

where $r = \|\boldsymbol{x}\|_2$.

The unbounded domain Ω^∞ is usually truncated in order to get a finite computational domain Ω. Then, Silver-Müller radiation conditions are approximated by an absorbing boundary condition (ABC), imposed in the artificial boundary (Givoli, 1992), or by using a PML (Berenger, 1994). In fact, it is standard to terminate a PML with a low-order ABC in order to provide better absorption of the outgoing waves, see Jin and Chew (1996). In this appendix a first-order approximation of the

Silver-Müller radiation conditions is detailed. In Appendix E two PMLs are discussed, optimized and compared in the context of high-order DG methods.

D.1.2 The Radar Cross Section

The radar cross section (RCS) is one of the most important quantities of interest in electromagnetic scattering problems. It provides a description of how an object reflects an incident electromagnetic wave (Harrington, 1961). For example, the *scattering width* is defined in 3D as

$$\chi(\phi,\theta) = \lim_{r\to\infty} 4\pi r^2 \frac{|E_3^S|^2}{|E_3^i|^2} = \lim_{r\to\infty} 4\pi r^2 \frac{|H_3^S|^2}{|H_3^i|^2}.$$

where ϕ and θ are the viewing angles, see Balanis (1989). In 2D the scattering with is defined as

$$\chi(\phi) = \lim_{r\to\infty} 2\pi r \frac{|H_3^S|^2}{|H_3^i|^2},$$

for the TE mode, and

$$\chi(\phi) = \lim_{r\to\infty} 2\pi r \frac{|E_3^S|^2}{|E_3^i|^2},$$

for the TM mode. Typically, the RCS (per unit area/length) is measured in Decibels, this conversion is achieved by computing

$$RCS = 10\log_{10}(\chi). \tag{D.13}$$

When only near field data is available, the RCS can be evaluated by performing a near-to-far field transformation, for further details see **?** or **?**Balanis (1989). For instance, the application of a near-to-far field transformation leads to the following alternative expression for the scattering width in 3D

$$\chi(\phi,\theta) = \frac{k^2}{4\pi}\Big(|L_\phi + N_\theta|^2 + |L_\theta - N_\phi|^2\Big), \tag{D.14}$$

where

$$N_\theta = \int_S \Big(J_1 \cos\theta\cos\phi + J_2 \cos\theta \sin\phi - J_3 \sin\theta \Big) e^{j\boldsymbol{k}\cdot\boldsymbol{x}'} dS',$$

$$N_\phi = \int_S \Big(- J_1 \sin\phi + J_2 \cos\phi \Big) e^{j\boldsymbol{k}\cdot\boldsymbol{x}'} dS',$$

$$L_\theta = \int_S \Big(M_1 \cos\theta\cos\phi + M_2 \cos\theta \sin\phi - M_3 \sin\theta \Big) e^{j\boldsymbol{k}\cdot\boldsymbol{x}'} dS',$$

$$L_\phi = \int_S \Big(- M_1 \sin\phi + M_2 \cos\phi \Big) e^{j\boldsymbol{k}\cdot\boldsymbol{x}'} dS',$$

S is a closed surface enclosing the scatterer, usually the scatterer surface, $j = \sqrt{-1}$, k is the wave number, $\boldsymbol{n}^S = (n_1^S, n_2^S, n_3^S)$ is the outward unit normal to S, $\mathbf{M} = -\boldsymbol{n}^S \times \boldsymbol{E}^f$ and $\mathbf{J} = \boldsymbol{n}^S \times \boldsymbol{H}^f$ are the electric and magnetic currents on S, respectively, and the superscript f indicates the complex amplitudes of the fields in the frequency domain.

In 2D, the expression of the scattering width (D.14) is simplified. For the TE mode the scattering width is computed as

$$\chi(\phi) = \frac{k}{4} \left| \int_C \Big((n_2^C \sin\phi + n_1^C \cos\phi) H_3 + \sqrt{\frac{\mu}{\varepsilon}} (n_2^C E_1^f - n_1^C E_2^f) \Big) e^{jw(x\cos\phi + y\sin\phi)} dl \right|^2,$$

and for the TM mode,

$$\chi(\phi) = \frac{k}{4} \left| \int_C \Big((n_2^C \sin\phi + n_1^C \cos\phi) E_3 + \sqrt{\frac{\varepsilon}{\mu}} (n_1^C H_2^f - n_2^C H_1^f) \Big) e^{jw(x\cos\phi + y\sin\phi)} dl \right|^2,$$

where C is a closed curve enclosing the scatterer and $\boldsymbol{n}^C = (n_1^C, n_2^C)$ is the outward unit normal to C.

The RCS defined in Equation (D.13) is usually referred as the *bistatic* RCS because, for a given incident field, the scattering width χ is measured at all viewing angles ϕ and θ. In industrial applications the *monostatic* RCS is also useful, which is defined as the bistatic RCS in the backscattered angle. Time-domain scattering codes usually offers the RCS output for a single angle of incidence and therefore, the

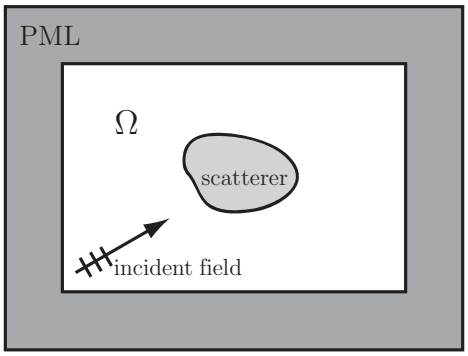


Figure D.1: Setup of an electromagnetic scattering problem with a PML surrounding the computational domain

computation of the monostatic RCS involves the simulation with different incident fields, being more expensive. A practical procedure to decide the number of angles needed for an accurate monostatic computation is described by Schuh et al. (1994). This reference recommends 36 computations for a monostatic computation with an scatterer of characteristic length equal to the wavelength λ. Moreover, a postprocess technique to enhance the quality of a monostatic computation is described. The idea is approximating the monostatic RCS in the vicinity of a computed angle using the bistatic RCS, see Schuh et al. (1994) for further details.

D.2 Discontinuous Galerkin formulation

This section reviews the DG discretization of a system of first-order hyperbolic equations, such as the Maxwell's curl equations written in conservative form (D.5). The computational domain is assumed bounded, for instance the truncation of the unbounded domain Ω^{∞}, considering an ABC or a PML surrounding the (bounded) computational domain Ω, see Figure D.1.

In the bounded domain Ω, a regular partition of elements $\overline{\Omega} = \bigcup_e \overline{\Omega_e}$ is assumed, such that $\Omega_i \bigcap \Omega_j = \emptyset$, for $i \neq j$. Then, an element by element discontinuous approximation space is considered, here, the space of polynomials of degree less or equal to

p in Ω_e, namely $\mathcal{P}^p(\Omega_e)$. The strong form (D.5) is multiplied by a vector of test functions $\boldsymbol{W} \in \left[\mathcal{P}^p(\Omega_e)\right]^{\mathtt{n_{comp}}}$, where $\mathtt{n_{comp}}$ is the number of components of $\boldsymbol{U}$ (for Maxwell equations, $\mathtt{n_{comp}}$=6 in 3D and $\mathtt{n_{comp}}$=3 in 2D). After integration in Ω_e, and integration by parts, the following equation is obtained

$$\int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{U}_e}{\partial t}\, d\Omega - \int_{\Omega_e} \frac{\partial \boldsymbol{W}}{\partial x_k} \cdot \boldsymbol{F}_k(\boldsymbol{U}_e)\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{W} \cdot \boldsymbol{F}_{\boldsymbol{n}}(\boldsymbol{U}_e)\, d\Gamma = \int_{\Omega_e} \boldsymbol{W} \cdot \boldsymbol{S}(\boldsymbol{U}_e)\, d\Omega,$$

where $\boldsymbol{U}_e$ denotes the restriction of $\boldsymbol{U}$ to the element Ω_e, $\boldsymbol{n}$ is the outward unit normal vector to $\partial\Omega_e$, and the normal flux is defined as

$$\boldsymbol{F}_n(\boldsymbol{U}) = \boldsymbol{F}_k(\boldsymbol{U}) n_k, \tag{D.15}$$

with n_k the kth component of $\boldsymbol{n}$. As standard in DG methods, in order to take into account the discontinuous nature of the approximation, the normal flux at the boundary of the element is replaced by a numerical normal flux, $\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}, \boldsymbol{U}^{\text{out}})$, which is evaluated in terms of the solution in the current element Ω_e and the solution at neighboring elements,

$$\boldsymbol{U}_e^{\text{out}}(\boldsymbol{x}) = \lim_{\tau \to 0^+} \boldsymbol{U}_e(\boldsymbol{x} + \tau \boldsymbol{n}) \quad \text{for } \boldsymbol{x} \in \partial\Omega_e. \tag{D.16}$$

Introducing the numerical flux, the DG weak form is given by

$$\int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{U}_e}{\partial t}\, d\Omega - \int_{\Omega_e} \frac{\partial \boldsymbol{W}}{\partial x_k} \cdot \boldsymbol{F}_k(\boldsymbol{U}_e)\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{W} \cdot \widetilde{\boldsymbol{F}}_n(\boldsymbol{U}_e, \boldsymbol{U}_e^{\text{out}})\, d\Gamma = \int_{\Omega_e} \boldsymbol{W} \cdot \boldsymbol{S}(\boldsymbol{U}_e)\, d\Omega. \tag{D.17}$$

Some conditions are required for the definition of a proper numerical flux function: it must be conservative, Lipstchiz and verify some consistency conditions, see Cockburn (2004). A natural choice for linear hyperbolic problems is the *flux splitting* technique described by Donea and Huerta (2005), which corresponds to an upwind

approximation, see for instance Hesthaven and Warburton (2002), Kabakian et al. (2004) and Chen et al. (2005). The normal flux (D.15) is decomposed into incoming flux (superscript $-$) and outgoing flux (superscript $+$)

$$\boldsymbol{F}_n(\boldsymbol{U}) = \boldsymbol{F}_n^-(\boldsymbol{U}) + \boldsymbol{F}_n^+(\boldsymbol{U}),$$

and the numerical flux is computed as

$$\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}, \boldsymbol{U}^{\text{out}}) = \boldsymbol{F}_n^+(\boldsymbol{U}) + \boldsymbol{F}_n^-(\boldsymbol{U}^{\text{out}}). \tag{D.18}$$

The incoming and outgoing fluxes corresponds to the information of the negative and positive eigenvalues of the Jacobian matrix $\mathbf{A}_{\boldsymbol{n}} = \frac{\partial \boldsymbol{F}_n}{\partial \boldsymbol{U}}$. That is,

$$\boldsymbol{F}_{\boldsymbol{n}}^-(\boldsymbol{U}) = \mathbf{A}_{\boldsymbol{n}}^- \boldsymbol{U}, \quad \boldsymbol{F}_{\boldsymbol{n}}^+(\boldsymbol{U}) = \mathbf{A}_{\boldsymbol{n}}^+ \boldsymbol{U},$$

with

$$\mathbf{A}_{\boldsymbol{n}}^+ = \tfrac{1}{2}\left(\mathbf{A}_{\boldsymbol{n}} + |\mathbf{A}_{\boldsymbol{n}}|\right), \qquad \mathbf{A}_{\boldsymbol{n}}^- = \tfrac{1}{2}\left(\mathbf{A}_{\boldsymbol{n}} - |\mathbf{A}_{\boldsymbol{n}}|\right).$$

Recall that if $\mathbf{A}_{\boldsymbol{n}} = \mathbf{X}\boldsymbol{\Lambda}\mathbf{X}^{-1}$ with $\boldsymbol{\Lambda}$ diagonal, then $|\mathbf{A}_{\boldsymbol{n}}| = \mathbf{X}|\boldsymbol{\Lambda}|\mathbf{X}^{-1}$.

After integration by parts of the second term in Equation (D.17), the following equivalent DG weak form is obtained,

$$\int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{U}_e}{\partial t}\, d\Omega + \int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{F}_k(\boldsymbol{U}_e)}{\partial x_k}\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{W} \cdot \left[\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}_e, \boldsymbol{U}_e^{\text{out}}) - \boldsymbol{F}_n(\boldsymbol{U}_e)\right] d\Gamma = \int_{\Omega_e} \boldsymbol{W} \cdot \boldsymbol{S}(\boldsymbol{U}_e)\, d\Omega. \tag{D.19}$$

Note that

$$\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}, \boldsymbol{U}^{\text{out}}) - \boldsymbol{F}_n(\boldsymbol{U}) = \boldsymbol{F}_n^-(\boldsymbol{U}^{\text{out}}) - \boldsymbol{F}_n^-(\boldsymbol{U}) = \mathbf{A}_n^- [\![\boldsymbol{U}]\!],$$

where the *jump operator* is defined as

$$[\![\boldsymbol{U}]\!] := \boldsymbol{U}^{\text{out}} - \boldsymbol{U}.$$

Thus, in that case, the weak form (D.19) can also be written as

$$\int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{U}_e}{\partial t}\, d\Omega + \int_{\Omega_e} \boldsymbol{W} \cdot \left(\frac{\partial \boldsymbol{F}_k(\boldsymbol{U}_e)}{\partial x_k} - \boldsymbol{S}(\boldsymbol{U}_e) \right) d\Omega + \int_{\partial\Omega_e} \boldsymbol{W} \cdot \mathbf{A}_n^- [\![\boldsymbol{U}_e]\!]\, d\Gamma = 0. \quad \text{(D.20)}$$

At an internal face/edge between two elements, the expression of the boundary term is

$$\mathbf{A}_n^- [\![\boldsymbol{U}]\!] = \frac{1}{2} \begin{pmatrix} -\boldsymbol{n} \times [\![\boldsymbol{H}]\!] + \sqrt{\frac{\varepsilon}{\mu}} \boldsymbol{n} \times (\boldsymbol{n} \times [\![\boldsymbol{E}]\!]) \\ \boldsymbol{n} \times [\![\boldsymbol{E}]\!] + \sqrt{\frac{\mu}{\varepsilon}} \boldsymbol{n} \times (\boldsymbol{n} \times [\![\boldsymbol{H}]\!]) \end{pmatrix}. \quad \text{(D.21)}$$

For the TE mode in 2D, the expression is

$$\mathbf{A}_n^- [\![\boldsymbol{U}]\!] = \frac{1}{2} \left[[\![H_3]\!] - \sqrt{\frac{\varepsilon}{\mu}} [\![\alpha]\!] \right] \begin{pmatrix} -n_2 \\ n_1 \\ -\sqrt{\frac{\mu}{\varepsilon}} \end{pmatrix}, \quad \text{(D.22)}$$

where $\alpha := n_1 E_2 - n_2 E_1$, and, for the TM mode

$$\mathbf{A}_n^- [\![\boldsymbol{U}]\!] = \frac{1}{2} \left[[\![E_3]\!] - \sqrt{\frac{\mu}{\varepsilon}} [\![\beta]\!] \right] \begin{pmatrix} n_2 \\ -n_1 \\ -\sqrt{\frac{\varepsilon}{\mu}} \end{pmatrix}, \quad \text{(D.23)}$$

where $\beta := n_2 H_1 - n_1 H_2$.

The DG weak form (D.20) is discretized in each element, leading to a system of ordinary differential equations (ODEs)

$$\mathbf{M} \frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \quad \text{(D.24)}$$

where $\mathbf{U}$ is the vector of nodal values (or approximation coefficients in a more general case), $\mathbf{M}$ is a block diagonal mass matrix and $\mathbf{R}(\mathbf{U})$ is the residual vector (see the implementation details in Section D.3).

The ODE system (D.24) is advanced in time using a standard explicit fourth-order Runge-Kutta scheme, see for instance Donea and Huerta (2005). The stability condition requires a time step

$$\Delta t \leq \widetilde{C}\frac{h}{cp^2},$$

where $\widetilde{C}$ is a constant, h is the minimum element size, c is the velocity of the electromagnetic wave, and p is the degree of the polynomial approximation. Note however, that for large-scale computations a low-storage RK scheme provides an important save in memory, see Sármány et al. (2007) and references therein.

The time marching process is stopped when the relative error in the scattering width between two consecutive cycles is small enough. A relative error in the RCS of order 10^{-3} has been previously reported as being sufficient for engineering purposes (Hesthaven and Warburton, 2002), and furthermore is better than achieved with other numerical schemes, see Hachemi et al. (2004).

D.2.1 Boundary conditions and interfaces

At an inter-element boundary, $\boldsymbol{U}^{\text{out}}$ is defined as the solution in the neighboring element, see Equation (D.16), but at an element boundary intersecting $\partial\Omega$, $\boldsymbol{U}^{\text{out}}$ is not totally determined from the boundary conditions. For instance, at a PEC boundary, the tangential component of the electric field is given by Equation (D.11), but the tangential component of the magnetic field remains undetermined in the expression of $\mathbf{A}_n^-[\![\boldsymbol{U}]\!]$, see Equation (D.21).

As usual in the numerical solution of hyperbolic problems, *Rankine-Hugoniot jump conditions* are applied in order to determine the unknown conditions, see LeVeque (1992) and Donea and Huerta (2005). Rankine-Hugoniot conditions for a system of

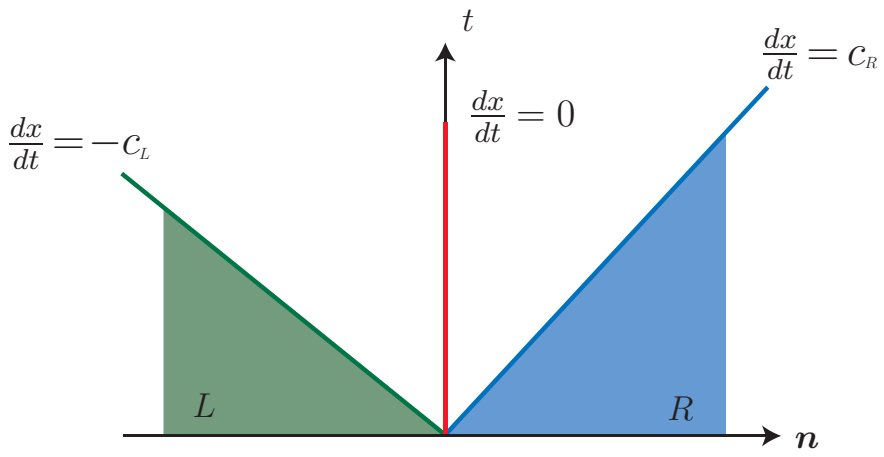


Figure D.2: Phase plane showing the characteristics for Maxwell equations

conservation laws such as (D.5) are

$$[\![\boldsymbol{F}_n]\!] = \lambda_j [\![\boldsymbol{U}]\!], \tag{D.25}$$

where λ_j are the eigenvalues of the Jacobian matrix $\mathbf{A}_n$, in 3D

$$\lambda_1 = \lambda_2 = -c_L, \qquad \lambda_3 = \lambda_4 = 0, \qquad \lambda_5 = \lambda_6 = c_R,$$

where $c_L = \dfrac{1}{\sqrt{\varepsilon_L \mu_L}}$ and $c_R = \dfrac{1}{\sqrt{\varepsilon_R \mu_R}}$ are the velocities of the electromagnetic wave in the corresponding media, L and R. Rankine-Hugoniot conditions (D.25) must be satisfied along the characteristics in the phase plane, see Figure D.2.

Material interface

At a material interface, only the tangential component of the electric field is prescribed, see Equations (D.7) and (D.8). Solving the linear system given by the Rankine-Hugoniot jump conditions (D.25) in 3D, the following expressions are ob-

tained for the tangential component of the fields on the interface

$$\boldsymbol{n} \times \boldsymbol{E}^{\text{out}} = \boldsymbol{n} \times \frac{(c_L \varepsilon_L \boldsymbol{E}^L - \boldsymbol{n} \times \boldsymbol{H}^L) + (c_R \varepsilon_R \boldsymbol{E}^R + \boldsymbol{n} \times \boldsymbol{H}^R)}{c_L \varepsilon_L + c_R \varepsilon_R}$$

$$\boldsymbol{n} \times \boldsymbol{H}^{\text{out}} = \boldsymbol{n} \times \frac{(c_L \mu_L \boldsymbol{H}^L + \boldsymbol{n} \times \boldsymbol{E}^L) + (c_R \mu_R \boldsymbol{H}^R - \boldsymbol{n} \times \boldsymbol{E}^R)}{c_L \mu_L + c_R \mu_R}. \tag{D.26}$$

In 2D, the application of Rankine-Hugoniot conditions lead to the following expressions

$$H_3^{\text{out}} = \frac{c_R \mu_R H_3^R + c_L \mu_L H_3^L - (\alpha^R - \alpha^L)}{c_R \mu_R + c_L \mu_L},$$

$$\alpha^{\text{out}} = \frac{c_R \varepsilon_R \alpha^R + c_L \varepsilon_L \alpha^L - (H_3^R - H_3^L)}{c_R \varepsilon_R + c_L \varepsilon_L}.$$

for the TE mode, and

$$E_3^{\text{out}} = \frac{c_R \varepsilon_R E_3^R + c_L \varepsilon_L E_3^L - (\beta^R - \beta^L)}{c_R \varepsilon_R + c_L \varepsilon_L},$$

$$\beta^{\text{out}} = \frac{c_R \mu_R \beta^R + c_L \mu_L \beta^L - (E_3^R - E_3^L)}{c_R \mu_R + c_L \mu_L},$$

for the TM mode.

Using the expressions of the tangential fields on the interface, the boundary term is evaluated using Equations (D.21), (D.22) or (D.23), where $[\![\boldsymbol{E}]\!] = \boldsymbol{E}^{\text{out}} - \boldsymbol{E}^L$ and $[\![\boldsymbol{H}]\!] = \boldsymbol{H}^{\text{out}} - \boldsymbol{H}^L$ for the left element, and $[\![\boldsymbol{E}]\!] = \boldsymbol{E}^{\text{out}} - \boldsymbol{E}^R$ and $[\![\boldsymbol{H}]\!] = \boldsymbol{H}^{\text{out}} - \boldsymbol{H}^R$ for the right element.

Perfect electric conductors

The boundary term for a PEC boundary is obtained particularizing the equations for a material interface in the case of a material of very high (electrical) conductivity. Note that the tangential component of $\boldsymbol{E}^{\text{out}}$ is known, see Equation (D.11). The

tangential components of $\boldsymbol{H}^{\text{out}}$ are obtained from Equation (D.26), that is,

$$\boldsymbol{n} \times \boldsymbol{H}^{\text{out}} = \boldsymbol{n} \times \boldsymbol{H} - \sqrt{\frac{\varepsilon}{\mu}} \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{E}^{\text{out}} - \boldsymbol{n} \times \boldsymbol{E}).$$

For the TE mode in 2D, α^{out} is known from the incident field, that is, $\alpha^{\text{out}} = -\alpha^{i}$, and the transverse field H_3 is computed as

$$H_3^{\text{out}} = H_3 - \sqrt{\frac{\varepsilon}{\mu}} [\![\alpha]\!].$$

For the TM mode in 2D, the transverse field E_3^{out} is known from the incident field and the tangential component of the magnetic field is determined as

$$\beta^{\text{out}} = \beta^{L} + \sqrt{\frac{\varepsilon}{\mu}} [\![E_3]\!].$$

Again, using the expressions of the tangential fields on the interface, the boundary term is evaluated using Equations (D.21), (D.22) or (D.23).

Far field boundary condition

The computational domain must be truncated and an artificial boundary condition is necessary to approximate the Silver-Müller radiation conditions (D.12). Here a PML is considered along the artificial boundary, see Appendix E. When a PML is introduced, a low-order ABC conditions is usually introduced at the outer boundary in order to provide better absorption of the outgoing waves (Jin and Chew, 1996). Here a first-order approximation of the Silver-Müller radiation conditions (D.12) is considered. This approach corresponds to impose null incoming flux at the outer boundary. Thus, the numerical flux (D.18) corresponds to

$$\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}, \boldsymbol{U}^{\text{out}}) = \boldsymbol{F}_n^{+}(\boldsymbol{U}) = \mathbf{A}_n^{+} \boldsymbol{U}.$$

or equivalently, the boundary term in the weak form (D.20) is given by

$$\mathbf{A}_n^- [\![\boldsymbol{U}]\!] = -\mathbf{A}_n^- \boldsymbol{U}.$$

It is worth noting that this first-order accurate ABC can be directly used, without a PML, provided that the outer boundary is placed far enough from the scatterer. This options is usually combined with coarsening of the mesh to effectively dissipate the waves before reaching the artificial boundary, see Kabakian et al. (2004).

D.3 Implementation details

In this section the computation of the residual vector of the DG semi-discrete system (D.24) is detailed, and the flowchart of the developed DG code for solving Maxwell's equations is presented.

D.3.1 Residual computation

In this section the computational of the residual vector of the semi-discrete system (D.24) is detailed. First, the efficient implementation for simplex elements with planar faces proposed by Hesthaven and Warburton (2002) is recalled. Then, a detailed discussion of the residual computation for curved elements is given, highlighting the differences between curved FEs considered in Appendix C, that is, isoparametric FEs, cartesian FEs, p-FEM and NEFEM.

For the sake of simplicity, the scalar convection equation is considered next,

$$u_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} u = 0.$$

The generalization for vector problems is straight-forward.

For the convection equation, $\boldsymbol{U} = u$ is a scalar unknown and the fluxes are

$$F_k(u) = a_k u, \quad k = 1, \ldots, \mathtt{n_{sd}}.$$

where $\boldsymbol{a}$ is the convection velocity.

The computation of the residual vector involve integration on element interiors to compute the contribution of the divergence (and the source term if present), and integration on element boundaries to compute the contribution of the boundary fluxes, that is

$$\mathbf{R}(\mathbf{U}) = \mathbf{F}^{\Omega_e} + \mathbf{F}^{\partial\Omega_e}. \tag{D.27}$$

If a nodal interpolation of the solution is considered,

$$u \simeq u^h = \sum_{j=1}^{\mathtt{n_{en}}} u_j N_j$$

the expression of the divergence term in Equation (D.27) is given by

$$F_i^{\Omega_e} := \sum_{k=1}^{\mathtt{n_{sd}}} a_k \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_{\Omega_e} N_i \frac{\partial N_j}{\partial x_k} \, d\Omega \right) u_j, \tag{D.28}$$

$$F_i^{\Gamma_e} := \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_{\partial\Omega_e} a_n^- N_i \, N_j \, d\Gamma \right) [\![u_j]\!], \tag{D.29}$$

Elements with planar faces

For elements with planar faces, a linear mapping is defined between the reference element and the physical element with planar faces, the isoparametric mapping given by the coordinates of the vertices of Ω_e. Then, integral (D.28) is computed in the reference element I as

$$F_i^{\Omega_e} = |J| \sum_{k=1}^{\mathtt{n_{sd}}} a_k \sum_{l=1}^{\mathtt{n_{sd}}} J_{l,k}^{-1} \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_I N_i \frac{\partial N_j}{\partial \xi_l} \, d\Omega \right) u_j,$$

where $|J|$ is the (constant) Jacobian of the linear mapping relating I and Ω_e. Thus, elemental matrices can be precomputed once for the reference element,

$$\mathbf{C}^{\xi_l} = \int_I N_i \frac{\partial N_j}{\partial \xi_l} \, d\Omega,$$

and used in the computation of integral (D.28) for each element. That is,

$$F_i^{\Omega_e} = |J| \sum_{k=1}^{\mathtt{n_{sd}}} a_k \sum_{l=1}^{\mathtt{n_{sd}}} J_{l,k}^{-1} \sum_{j=1}^{\mathtt{n_{en}}} C_{ij}^{\xi_l} \, u_j,$$

The computation of the flux term (D.29) for planar faces is performed using the same idea. A linear mapping between a reference face Γ_I and the physical planar face Γ_e is considered. Then, integral (D.29) is computed as

$$F_i^{\Gamma_e} = a_n^- \sum_{j=1}^{\mathtt{n_{fn}}} \left(\int_{\Gamma_I} N_i \, N_j \, d\Gamma \right) [\![u_j]\!].$$

Note that, in a planar face, a_n^- is constant due to the unique outward unit normal to planar face. Note also that, the sum over element nodes is replaced by a sum over the number of face nodes, $\mathtt{n_{fn}}$, because other shape functions over the complete planar face.

Thus the computation of the flux term for each planar face is carried out as

$$F_i^{\Gamma_e} = a_n^- \sum_{j=1}^{\mathtt{n_{fn}}} m_{ij} [\![u_j]\!]$$

where

$$m_{ij} = \int_{\Gamma_I} N_i \, N_j \, d\Gamma$$

is the mass matrix for the reference face, computed only once, and used in the computation of the flux term for each physical face.

Note that the linearity of Maxwell's equations (constant $\mathbf{A}_n^-$) and the use of tri-

angular or tetrahedral meshes (constant Jacobian) allows an efficient implementation of the DG method. This is usually referred as the *quadrature free* implementation of DG methods (Atkins and Shu, 1998).

Curved elements

For curved elements, the computation of integral (D.28) is different depending on the curved FE technique considered, see Appendix C.

When the interpolation is defined in the reference element (isoparametric FEs and p-FEM), the element integral (D.28) is computed as

$$F_i^{\Omega_e} = \sum_{k=1}^{\mathtt{n_{sd}}} a_k \sum_{l=1}^{\mathtt{n_{sd}}} \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_I N_i J_{l,k}^{-1} \frac{\partial N_j}{\partial \xi_l} |J| \, d\Omega \right) u_j,$$

where $|J|$ is the Jacobian of the non-linear mapping relating local and cartesian coordinates (isoparametric mapping for isoparametric FEs and an exact mapping for p-FEM). If a cartesian interpolation is considered (cartesian FEs and NEFEM), the element integral (D.28) is computed as described in Appendix C. In any case, for curved elements it is necessary to perform numerical integration for each element.

Similarly, when the interpolation is defined in a reference element, integral of the flux term (D.29) is computed as

$$F_i^{\Gamma_e} = \sum_{j=1}^{\mathtt{n_{fn}}} \left(\int_{\Gamma_I} a_n^- N_i \; N_j |J| \, d\Gamma \right) [\![u_j]\!],$$

and, when the interpolation is directly defined in cartesian coordinates, the flux term is computed as

$$F_i^{\Gamma_e} = \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_{\Omega_{eh}} a_n^- N_i \; N_j \, d\Gamma \right) [\![u_j]\!],$$

for cartesian FEs, and

$$F_i^{\Gamma_e} = \sum_{j=1}^{\mathtt{n_{en}}} \left(\int_{\Omega_e} a_n^- N_i \; N_j \, d\Gamma \right) [\![u_j]\!],$$

for NEFEM. Note that the difference between cartesian FEs and NEFEM is in the domain of integration. For cartesian FEs, the integral is computed over the (approximated) computational element Ω_{eh} whereas NEFEM considers the exact geometry of the physical element Ω_e, see Appendix C. Note also that the sum is defined over $\mathtt{n_{en}}$ for cartesian approximations, whereas the sum is defined over $\mathtt{n_{fn}}$ if the interpolation is defined in local coordinates, see the discussion of Section B.1.

D.3.2 Developing a code for solving Maxwell's equations

The flow chart of the developed code for solving Maxwell's equations with a DG formulation is depicted in Figure D.3.

The computation starts with some geometric definitions such as the tetrahedral mesh, the information of the reference element and the information of faces that are classified into interiors, PEC or ABC. The geometric information in blue, parametric coordinates of boundary nodes, is only necessary for methods with an exact boundary representation, that is for p-FEM and NEFEM.

Before starting the time integration process elemental matrices for the reference element and reference face are precomputed. The initial condition is loaded (usually zero fields everywhere) and the time marching process starts. For each time step, the computation of the residual vector is performed with a loop on faces to compute the flux term (D.29), and a loop on elements to compute the divergence term (D.28). Alternatively, it is possible to implement a single loop on elements and a loop on element faces inside, but note that the proposed implementation is more efficient. For a given face the contribution of (D.28) to the residual vector is computed for left and right elements at the same time.

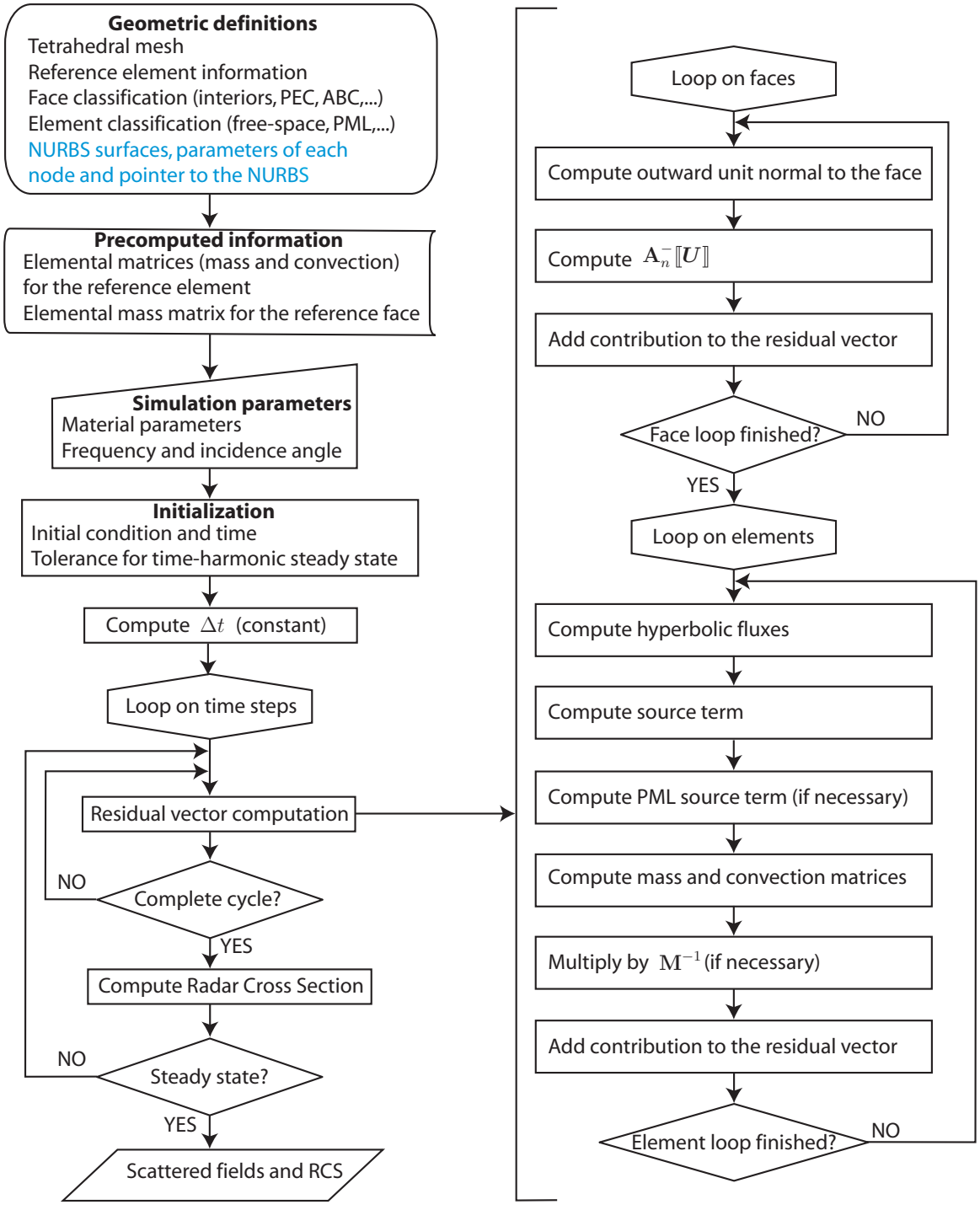


Figure D.3: Flowchart of the developed code for solving Maxwell's equations with a DG formulation

For each face, the outward unit normal $\boldsymbol{n}$ is computed as

$$\boldsymbol{n} = \frac{1}{\|\mathbf{t}^{\mathbf{F}}\|}(t_2^F, -t_1^F), \qquad \boldsymbol{n} = \frac{\mathbf{t}^F \times \widehat{\mathbf{t}}^F}{\|\mathbf{t}^F \times \widehat{\mathbf{t}}^F\|},$$

in 2D and 3D respectively, where $\mathbf{t}^F$ and $\widehat{\mathbf{t}}^F$ define the tangent directions to the face.

For planar faces $\mathbf{t}^F = \boldsymbol{x}_2 - \boldsymbol{x}_1$ and $\widehat{\mathbf{t}}^F = \boldsymbol{x}_3 - \boldsymbol{x}_1$, where $\boldsymbol{x}_k$, $k = 1 \ldots \mathtt{n_{sd}}$, are the coordinates of the face vertices. For curved faces with an isoparametric (approximated) boundary representation

$$\mathbf{t}^F = \left(\sum_{j=1}^{\mathtt{n_{fn}}} x_j \frac{\partial N_j}{\partial \xi}, \sum_{j=1}^{\mathtt{n_{fn}}} y_j \frac{\partial N_j}{\partial \xi}, \sum_{j=1}^{\mathtt{n_{fn}}} z_j \frac{\partial N_j}{\partial \xi} \right),$$

and

$$\widehat{\mathbf{t}}^F = \left(\sum_{j=1}^{\mathtt{n_{fn}}} x_j \frac{\partial N_j}{\partial \eta}, \sum_{j=1}^{\mathtt{n_{fn}}} y_j \frac{\partial N_j}{\partial \eta}, \sum_{j=1}^{\mathtt{n_{fn}}} z_j \frac{\partial N_j}{\partial \eta} \right),$$

where (x_j, y_j, z_j) are the coordinates of the $\mathtt{n_{fn}}$ face nodes. Finally, when an exact boundary representation is considered, tangent directions are computed using the NURBS parametrization. In 2D, $\mathbf{t}^F = \boldsymbol{C}'$ and, in 3D, $\mathbf{t}^F = \dfrac{\partial \boldsymbol{S}}{\partial \lambda}$ and $\widehat{\mathbf{t}}^F = \dfrac{\partial \boldsymbol{S}}{\partial \kappa}$.

The computation of the boundary term $\mathbf{A}_n^- [\![U]\!]$ is detailed in the previous section. Different expressions are used for internal faces, PEC faces, boundary faces or material interfaces. The face classification in the first step of the flow chart allows to identify the expression of $\mathbf{A}_n^- [\![U]\!]$ to be used in each face. Then, the contribution of the flux (boundary) term is added to the residual vector. Recall that using a cartesian interpolation all positions of the residual vector corresponding to element nodes are affected, whereas with an interpolation defined in local coordinates only positions corresponding to face nodes are modified.

Next, the contribution of the divergence term is computed with a loop on elements. The expression of the hyperbolic fluxes corresponding to the conservative form of Maxwell's equations is given in Section D.1. When a PML is used, see next Appendix, an extra source term is computed for elements lying in the absorbing region. The

computation of mass and convection matrices is different for elements with planar faces and curved elements, as detailed in Section D.3.1. Finally, if needed, the residual vector is multiplied by the inverse of the mass matrix. Note that the mass matrix is block diagonal and, therefore, the multiplication can be done inside the element loop, element by element. In addition, for planar element, the inverse of the mass matrix of the reference element can be precomputed and used for each element.

After each time step, if a complete cycle is performed, the RCS is computed and compared with the RCS of the previous cycle. If the relative difference is small enough the time-harmonic steady state is reached and the computation finishes. In fact, the stop criteria is checked over the scattering width to avoid slow convergence when singularities associated to the logarithmic scale of the RCS are present. The outputs of the computation are the scattered fields and the RCS.

Appendix E

Comparison of two Perfectly Matched Layers

The numerical simulation of electromagnetic scattering problems involves approximating the interaction between a known incident field and a scatterer. The scattered field produced by this interaction is a wave that propagates outwards towards infinity. There are many possible approaches to the numerical solution of this problem, reference Givoli (1992) presents an overview. Of these approaches, methods based on boundary integral equations (Amini and Kirkup, 1995), infinite elements (Bettess, 1992), non-reflecting boundary conditions (Givoli, 1992), and absorbing layers are widely used. In particular, methods based on absorbing layers involve truncation of the infinite domain, and absorbing the outgoing scattered waves at the truncated boundary through the addition of an artificial layer of material surrounding the computational domain. The absorbing layer known as the perfectly matched layer (PML), in which the waves are absorbed exponentially without reflection, see Berenger (1994), Abarbanel and Gottlieb (1998) and Turkel and Yefet (1998), is popular in the CEM community.

The first PML was introduced by Berenger (1994) to solve Maxwell's equations in the time-domain using a finite difference (FD) scheme. A splitting of the electric and magnetic fields is undertaken in an extra layer surrounding the computational domain, and, within this layer, an artificial conductivity is introduced in order to absorb the outgoing waves. The success of this method motivated several enhancements, and

the extension to other numerical methods, such as the FEM, see Polycarpou et al. (1996), or the FV method, see Bonnet and Poupaud (1997). From a computational point of view, the main disadvantage of the Berenger PML is the increase on computational memory requirements due to field splitting, which can be considerable for three dimensional large-scale problems. Abarbanel et al. (2006) propose a new PML in which no splitting of the fields is undertaken. Therefore, it is a good alternative for large-scale simulations. To the best of our knowledge no previous experience has been reported with this novel approach. This is a considerable oversight as numerical experimentation is crucial to obtain the best performance of a PML.

This appendix presents a numerical study to select the optimal parameters for both the Berenger and the non-linear PMLs in the context of high-order DG methods. A comparison is presented in terms of accuracy, and some comments are given about memory requirements.

E.1 Perfectly Matched Layers (PMLs)

This section is devoted to recall the classical Berenger PML proposed by Berenger (1994) and the non-linear PML recently proposed by Abarbanel et al. (2006).

In order to introduce the PML, the computational domain is first enlarged. This layer, which makes up the PML, is defined by $\Omega^{\mathrm{PML}} = \bigcup_{i=1}^{3} \Omega_i^{\mathrm{PML}}$, where

$$\Omega_1^{\mathrm{PML}} = \{(x,y); x \in [-A,-a] \cup [a,A], y \in [-b,b]\}$$

$$\Omega_2^{\mathrm{PML}} = \{(x,y); x \in [-a,a], y \in [-B,-b] \cup [b,B]\}$$

$$\Omega_3^{\mathrm{PML}} = \{(x,y); x \in [-A,-a] \cup [a,A], y \in [-B,-b] \cup [b,B]\},$$

and as illustrated in Figure E.1. Note that the computational domain Ω is assumed to be with rectangular exterior boundary $[-a,a] \times [-b,b]$.

In this layer a non-physical conductivity $\sigma = \sigma_1 + \sigma_2$ is introduced in order to

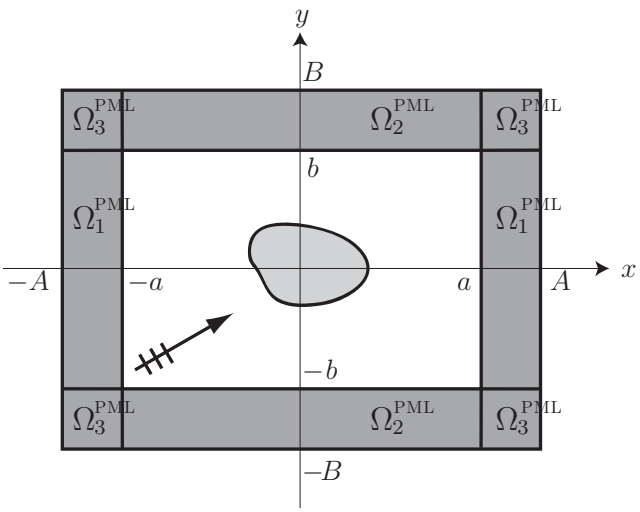


Figure E.1: Design of a two dimensional PML

absorb the outgoing waves. For a given point $\boldsymbol{x} = (x, y)$, the conductivity is defined as

$$\sigma_1 = \begin{cases} \sigma_0 \Big(\dfrac{x-a}{A-a} \Big)^n & \text{if} \quad \boldsymbol{x} \in \Omega_1^{\text{PML}} \cup \Omega_3^{\text{PML}} \\ 0 & \text{otherwise} , \end{cases}$$

and

$$\sigma_2 = \begin{cases} \sigma_0 \Big(\dfrac{y-b}{B-b} \Big)^n & \text{if} \quad \boldsymbol{x} \in \Omega_2^{\text{PML}} \cup \Omega_3^{\text{PML}} \\ 0 & \text{otherwise} . \end{cases}$$

It is important to note that the non-physical conductivity is assumed to have a polynomial profile in the absorbing layer, being zero in free-space. The parameter n is the degree of absorption and σ_0 is the maximum conductivity in the absorbing layer. The PML region is usually terminated using a zero tangential electric field condition ($\boldsymbol{n} \times \boldsymbol{E} = \boldsymbol{0}$) or a characteristic boundary condition. In this work a characteristic boundary condition is used throughout, which corresponds to a first-order approximation of the Silver-Müller radiation condition, see Appendix D.2.1. Nevertheless, it is worth remarking that when the simulations reported here were repeated with a zero tangential electric field condition no substantial differences were obtained. However, important differences between these two alternatives are observed if the PML

thickness is reduced, see Jin and Chew (1996) for further details.

In the following, the 2D TM mode, given by Equations (D.5) and (D.6) in Appendix D.1, is considered as a model problem. Both the Berenger and the non-linear PMLs are reviewed, and some comments are given about their implementation in a DG framework.

E.1.1 The Berenger PML

Following Bonnet and Poupaud (1997), Maxwell system augmented with a Berenger PML can be written in conservative form as

$$\frac{\partial \hat{\boldsymbol{U}}}{\partial t} + \frac{\partial \hat{\boldsymbol{F}}_k(\hat{\boldsymbol{U}})}{\partial x_k} = \hat{\boldsymbol{S}}(\hat{\boldsymbol{U}}) + \boldsymbol{S}^B(\hat{\boldsymbol{U}}),$$

where

$$\hat{\boldsymbol{U}} = \begin{pmatrix} \boldsymbol{U} \\ \varepsilon E_3^y \end{pmatrix}, \quad \hat{\boldsymbol{F}}_1 = \begin{pmatrix} \boldsymbol{F}_1 \\ 0 \end{pmatrix}, \quad \hat{\boldsymbol{F}}_2 = \begin{pmatrix} \boldsymbol{F}_2 \\ H_1 \end{pmatrix}, \quad \hat{\boldsymbol{S}} = \begin{pmatrix} \boldsymbol{S} \\ 0 \end{pmatrix},$$

$$\boldsymbol{S}^B = \begin{pmatrix} -\sigma_2 H_1 \\ -\sigma_1 H_2 \\ (\sigma_1 - \sigma_2) E_3^y - \sigma_1 E_3 \\ -\sigma_2 E_3^y \end{pmatrix},$$

and $\boldsymbol{U}$, $\boldsymbol{F}_k$ and $\boldsymbol{S}$ are defined in Equation (D.6) in Appendix D.1.

Note that the transverse field is split as $E_3 = E_3^x + E_3^y$. The first three equations are exactly the same as the classical Maxwell system with an extra source term. The fourth equation introduces a non-physical variable E_3^y that should be stored and solved in order to obtain the numerical solution of the complete system.

If a DG formulation is adopted, a new numerical flux must be deduced for the fourth equation. For example, for the two dimensional TM mode, the expression of the new numerical flux for an interior face in the PML region is

$$\widetilde{f}_n = \frac{1}{2}(H_1^{out} + H_1)n_2 - \frac{c}{2}[\![E_3]\!]n_2^2,$$

see Bonnet and Poupaud (1997) for further details.

E.1.2 The non-linear PML

The non-linear PML proposed by Abarbanel et al. (2006) can be written as the standard Maxwell system augmented with a non-linear source term

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_k(\boldsymbol{U})}{\partial x_k} = \boldsymbol{S}(\boldsymbol{U}) + \boldsymbol{S}^N(\boldsymbol{U}),$$

where $\boldsymbol{U}$, $\boldsymbol{F}_k$ and $\boldsymbol{S}$ are defined in Equation (D.6) in Appendix D.1. For the 2D TM mode the non-linear source term is given by

$$\boldsymbol{S}^N = \begin{pmatrix} -\sigma_2 H_1 \\ -\sigma_1 H_2 \\ -\dfrac{E_3}{M^2}(\sigma_1 H_2^2 - \sigma_2 H_1^2) \end{pmatrix},$$

where $M^2 = \iota(H_1^2 + H_2^2) + (1-\iota)E_3^2$, and ι is a free parameter. As noted by Abarbanel et al. (2006), the source term is not well defined when $\boldsymbol{E} = \boldsymbol{H} = \boldsymbol{0}$. This could possibly occur if the initial conditions correspond to zero fields or if complete absorption of the field takes place within the PML layer. To overcome this problem it is convenient to define $M^2 = \iota(H_1^2 + H_2^2) + (1-\iota)E_3^2 + \varsigma$ where ς is a regularization parameter, which is usually chosen to be lower than the truncation error. Moreover, in Abarbanel et al. (2006) the value of $\iota = 1/2$ is recommended.

In all the numerical examples presented in this work the parameters $\iota = 1/2$ and $\varsigma = 10^{-6}$ have been adopted for the non-linear PML. Through numerical experimentation it was found that no better performance could be obtained if other parameters are adopted. For example, using $\varsigma = 10^{-12}$ produces results which are practically indistinguishable form those obtained with $\varsigma = 10^{-6}$. However, using $\iota = 1$ or $\iota = 0$ sometimes lead to results that are not as accurate as those obtained by using $\iota = 1/2$.

Remark 4. *The non-linear character of the source term introduced in the PML region*

is not a disadvantage when an explicit time marching is used, but special attention should be paid to this property if an implicit time integration is preferred, see Persson and Peraire (2006a).

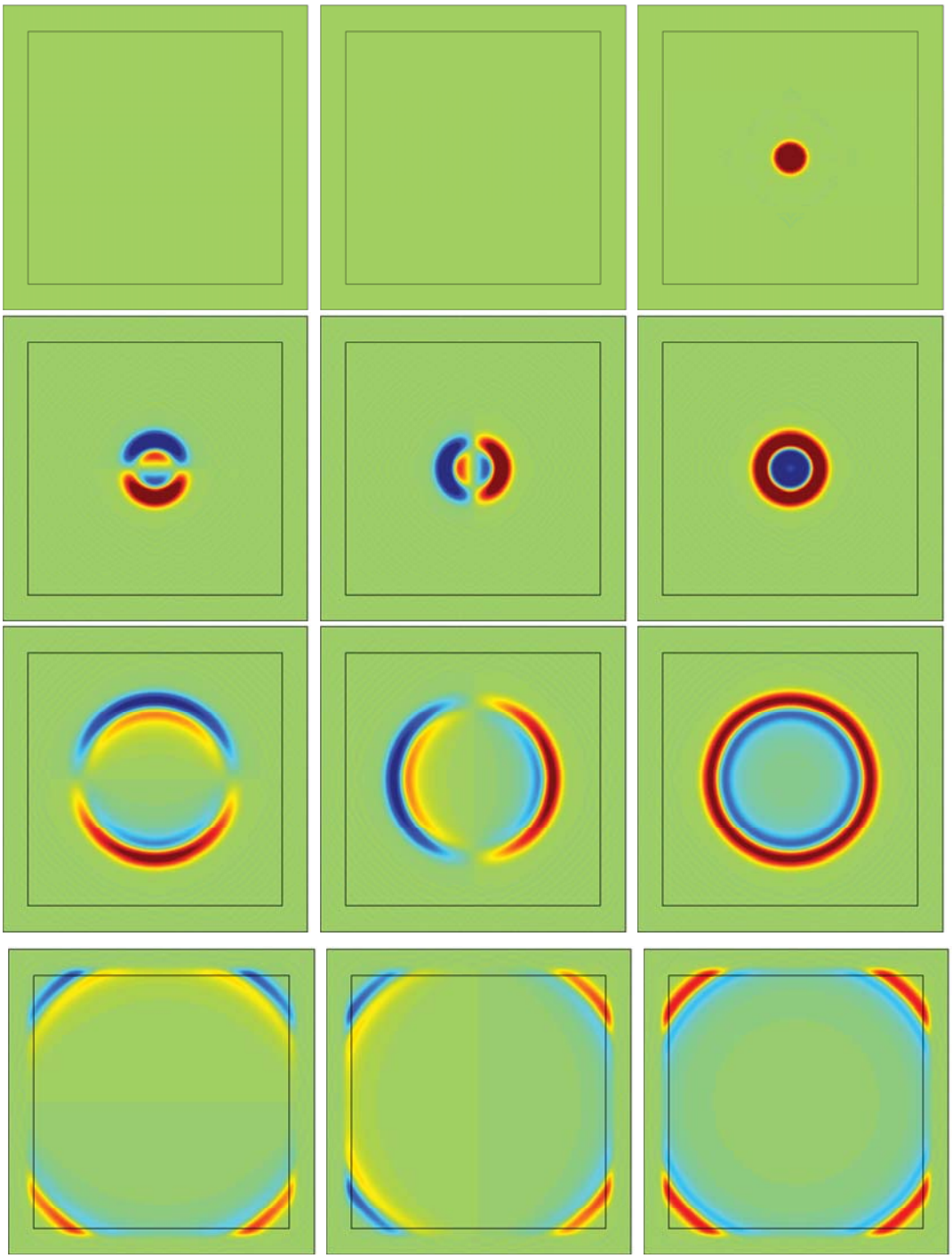
E.2 Validation

The numerical test considered for the validation of the PML is the propagation of a transient pulse in the 2D square domain $[-50, 50]^2$, see Abarbanel et al. (2006). The computational domain is surrounded by a PML of thickness 10 and it is terminated by a characteristic boundary condition. The initial condition is $H_1(x, y, 0) = H_2(x, y, 0) = 0$, and

$$E_3(x, y, 0) = e^{-\ln 2\left(\frac{x^2+y^2}{9}\right)}. \tag{E.1}$$

Figure E.2 shows the electric and magnetic fields at different times. The initial condition of Equation (E.1) generates a transient pulse that propagates from the origin outwards toward the infinity. When the pulse enters in the PML region an efficient absorption of the waves is observed. To measure absorption accuracy, a reference solution computed over a much larger domain is considered. The larger domain is defined as $[-350, 350]^2$ in order to ensure that no outgoing waves are reflected back into the domain at time $t = 250$. The numerical solution is compared with the reference solution over the line $\Gamma^{\mathtt{err}} = \{(x, y) \mid x = -40,\ y \in [-40, 40]\}$. Following Abarbanel et al. (2006), the time evolution of the $\mathcal{L}^2(\Gamma^{\mathtt{err}})$ error is represented in Figure E.3 for a polynomial approximation with $p = 4$. The results are very close to the results for the non-linear PML presented in Abarbanel et al. (2006) with a fourth-order FD scheme.

A comparison of the Berenger and the non-linear PMLs reveals important differences in accuracy, but recall that, for a given spatial discretization, the non-linear PML is more computationally efficient and it requires less memory than the Berenger PML, see the comparison in Section E.4. Finally, it is worth to remark that the PML parameters for this example are extracted from Abarbanel et al. (2006) to compare



(a) H_1 (b) H_2 (c) E_3

Figure E.2: Propagation of a pulse in a square domain: snapshots of the numerical solution at different times

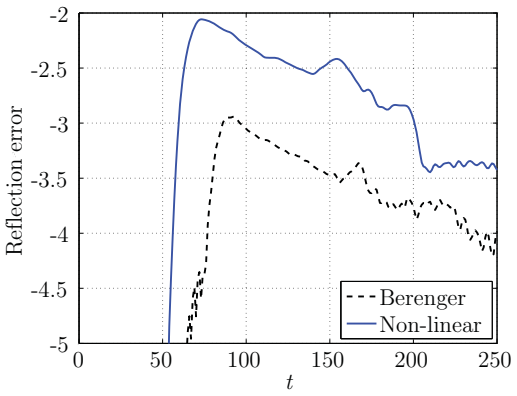


Figure E.3: Propagation of a pulse in a square domain: time evolution of the $\mathcal{L}^2(\Gamma^{\mathtt{err}})$ error for Berenger and non-linear PMLs

and validate the results, but the best performance of a PML requires an optimization of some parameters, see the next section.

E.3 Optimization of parameters

The design of an efficient PML, which minimizes reflections back in to the computational domain, requires the selection of several parameters such as the thickness of the PML region, the degree of absorption, and the maximum conductivity inside the PML, see Appendix E.1. For the classical Berenger PML, several papers can be found with different recommendations about these parameters. Nevertheless, the choice of optimal PML parameters depends on the numerical method considered, and, therefore, different recommendations can be found in a FV context (Bonnet and Poupaud, 1997), in a low-order FE context (Hachemi et al., 2004) or in FD methods (Taflove, 1995). For the non-linear PML no previous experience is available. This novel absorbing layer was proposed in Abarbanel et al. (2006) but no effort was made to investigate optimal parameters.

In this section a numerical study is presented in order to determine the optimal parameters for both the Berenger and the non-linear PMLs, in the context of high-order DG methods. The numerical test considers the TM mode where an incident

plane wave travelling in the x^+ direction is scattered by a PEC cylinder. To avoid errors caused by a non-exact representation of the curved boundary the NURBS-enhanced finite element method (NEFEM) is adopted. In all the computations the solution is advanced in time using an explicit fourth-order Runge-Kutta scheme until the time-harmonic steady state is reached.

Following Taflove (1995) the maximum conductivity in the PML region is defined in terms of the reflection error as

$$\sigma_0 = \frac{n+1}{2d} R,$$

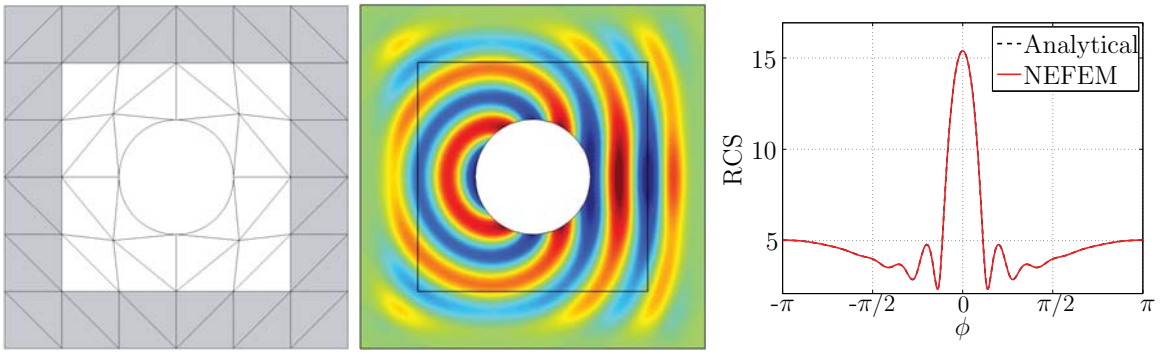
where n is the degree of absorption, d is the layer thickness, and the reflection error is e^{-R}. In the numerical experiments in this section the degree of absorption n is varied from 0 to 4, the parameter R is increased from 0 to 20 in increments of 2, d is taken to be λ and 2λ, where λ denotes the wavelength of the incident field, and the minimum distance between the scatterer and the PML is fixed as λ.

For a series of PEC cylinders at different frequencies, the error in the RCS is computed using the exact RCS of the PEC cylinder, see Harrington (1961) or Balanis (1989). In each case, the error associated with the converged DG solution for different PML parameters n, R and d is computed.

The first test case consists of a cylinder of diameter 2λ. A coarse mesh with only four elements for the discretization of the curved boundary is considered and high-order approximations are used to properly capture the solution. Figure E.4 shows the computational mesh, transverse scattered field E_3, and the RCS. The solution is computed using a degree of interpolation $p = 8$ and the non-linear PML of thickness λ with optimized parameters $R = 8$ and $n = 2$.

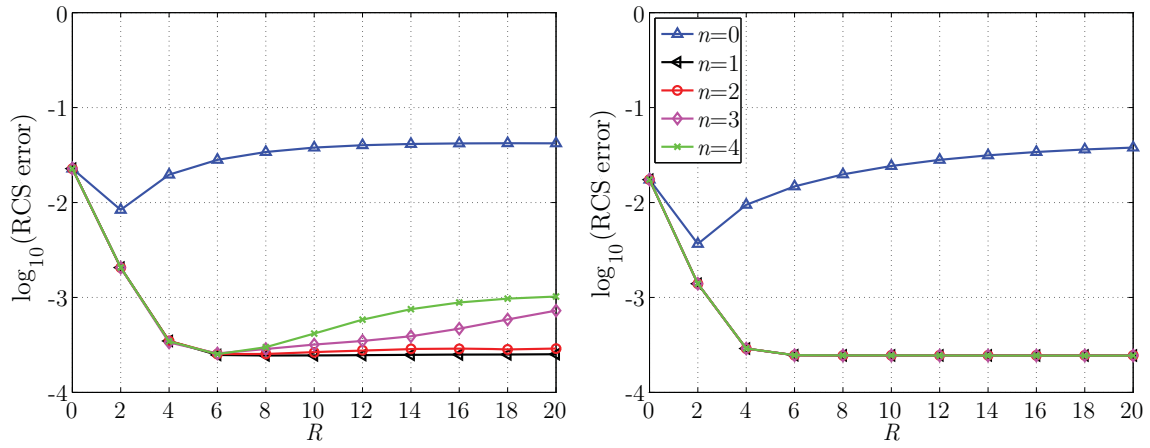
Figure E.5 shows the errors in the RCS, associated with the Berenger PML for a cylinder of diameter 2λ and $p = 8$, as a function of the parameter R. In this figure, the two plots show the errors associated with PMLs of thickness λ and 2λ, and on these plots the different line types are associated with different values of n.

When a thickness of λ is considered, the best performance of the PML is achieved



(a) Mesh (b) E_3 field (c) RCS

Figure E.4: Scattering by a PEC cylinder of diameter 2λ with optimal PML parameters



(a) Thickness λ (b) Thickness 2λ

Figure E.5: Berenger PML: error in the RCS as a function of R and n for the cylinder of diameter 2λ, with a degree of interpolation $p = 8$

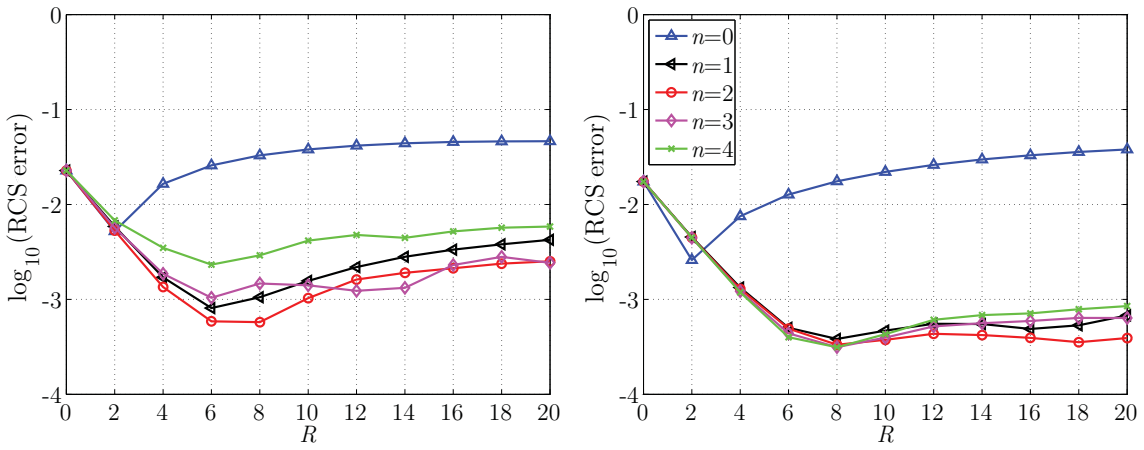
using a linear or quadratic degree of absorption. In this case, not only the lowest error is obtained, but also the error is independent of R if a sufficiently large value is considered. For $n > 2$, RCS error is highly dependent on the value of the R parameter (i.e. on the maximum conductivity in the PML). For instance, with $n = 4$ and $R = 6$ the PML performs three times better than using $n = 4$ and $R = 20$. Zero degree of absorption (corresponding to a constant value of the conductivity in the PML) gives

the worst results in all the experiments, with an error two orders of magnitude higher than using $n > 0$. In this case, the discontinuity in the material parameters between free-space region and the absorbing layer produces spurious reflections deteriorating the numerical solution near the scatterer. When the thickness is increased to 2λ, the error obtained when $n > 0$ and $R \geq 6$ is independent of R and n, see Figure E.5. This is an important feature of this PML, that is, an efficient absorption of the outgoing waves for an extensive range of its parameters.

Figure E.6 shows the errors in the RCS, associated with the non-linear PML for a cylinder of diameter 2λ and $p = 8$, as a function of the parameter R. With a layer of thickness λ the RCS error shows a strong dependence on the PML parameters. The lowest error and consequently the best performance of the non-linear PML is observed for a quadratic degree of absorption and a value of $R = 6$ or $R = 8$. Other choices may represent an important loss of accuracy. For instance, the RCS error with $n = 2$ and $R = 8$ is one order of magnitude lower than using $n = 4$ and $R = 16$. Important differences are found when a PML of thickness 2λ is considered. First, an optimal choice of the PML parameters provides lower errors than those obtained with thickness λ. This fact indicates that PML error dominates over the discretization error in the computations with a layer of thickness λ. Moreover, a weaker dependence of the RCS error on the PML parameters is observed when a sufficiently large value of R is considered, as was previously found with the Berenger PML.

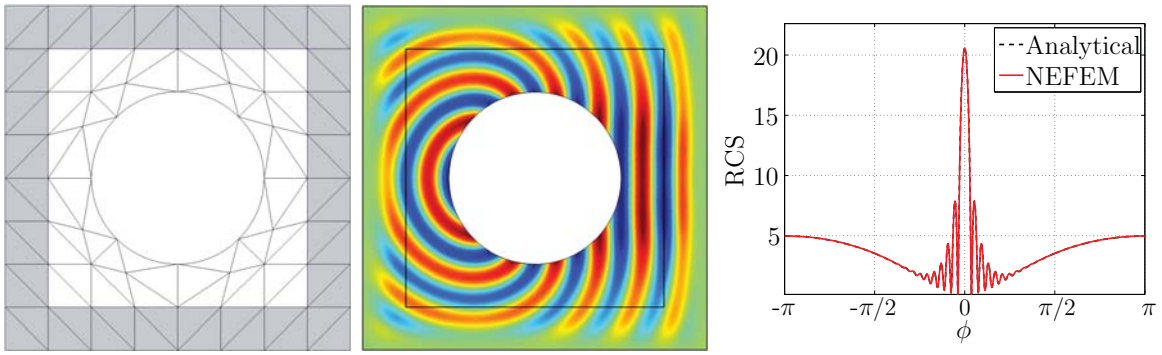
The second test case consists of a cylinder of diameter 4λ, which is used in order to investigate how the choice of the parameters R, d and n affect the performance of the Berenger and non-linear PMLs at a higher frequency. Figure E.7 shows the computational mesh, transverse scattered field E_3, and the RCS. The solution is computed using a degree of interpolation $p = 8$ and the non-linear PML of thickness λ with optimized parameters $R = 8$ and $n = 2$.

Figure E.8 shows the errors in the RCS, associated with the Berenger PML for a cylinder of diameter 4λ and $p = 8$, as a function of the parameter R. The best performance of the PML is obtained with a value of $R \geq 4$, for both linear or quadratic degree of absorption. A thickness of 2λ offers the same performance with a weaker



(a) Thickness λ (b) Thickness 2λ

Figure E.6: Non-linear PML: error in the RCS as a function of R and n for the cylinder of diameter 2λ, with a degree of interpolation $p = 8$

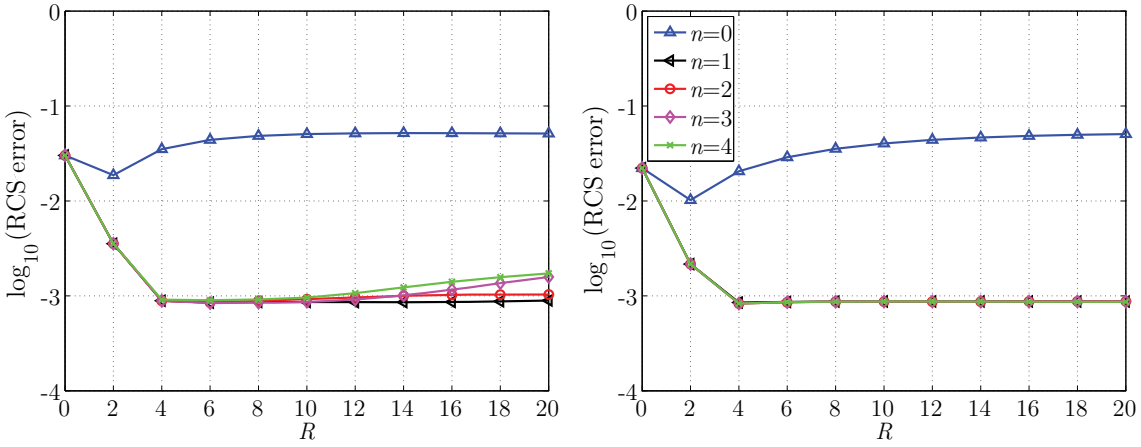


(a) Mesh (b) E_3 field (c) RCS

Figure E.7: Scattering by a PEC cylinder of diameter 4λ with optimal PML parameters

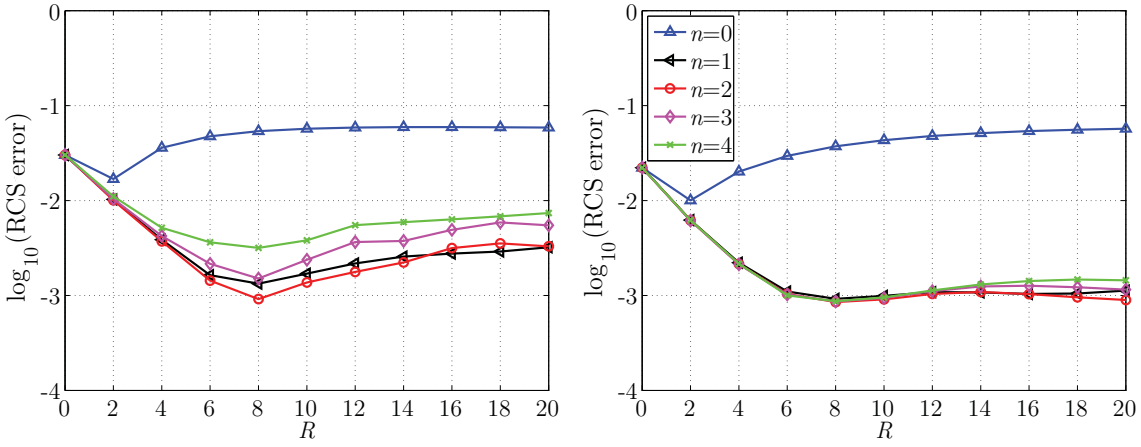
dependence on the PML parameters.

Figure E.9 shows the errors in the RCS, associated with the non-linear PML for a cylinder of diameter 4λ, as a function of the parameter R. When a thickness of λ is considered, the best performance of the PML is obtained with a quadratic degree of absorption. This choice slightly outperforms the results with a linear absorption profile. A value of $R = 8$ is optimal in this example. When the thickness of the



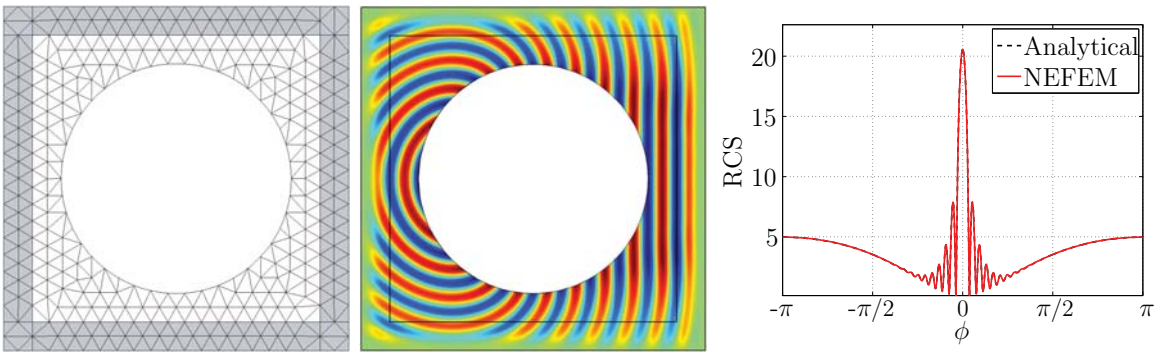
(a) Thickness λ (b) Thickness 2λ

Figure E.8: Berenger PML: error in the RCS as a function of R and n for the cylinder of diameter 4λ, with a degree of interpolation $p = 8$



(a) Thickness λ (b) Thickness 2λ

Figure E.9: Non-linear PML: error in the RCS as a function of R and n for the cylinder of diameter 4λ. Absorbing layer of thickness λ (left) and 2λ (right), with a degree of interpolation $p = 8$



(a) Mesh (b) E_3 field (c) RCS

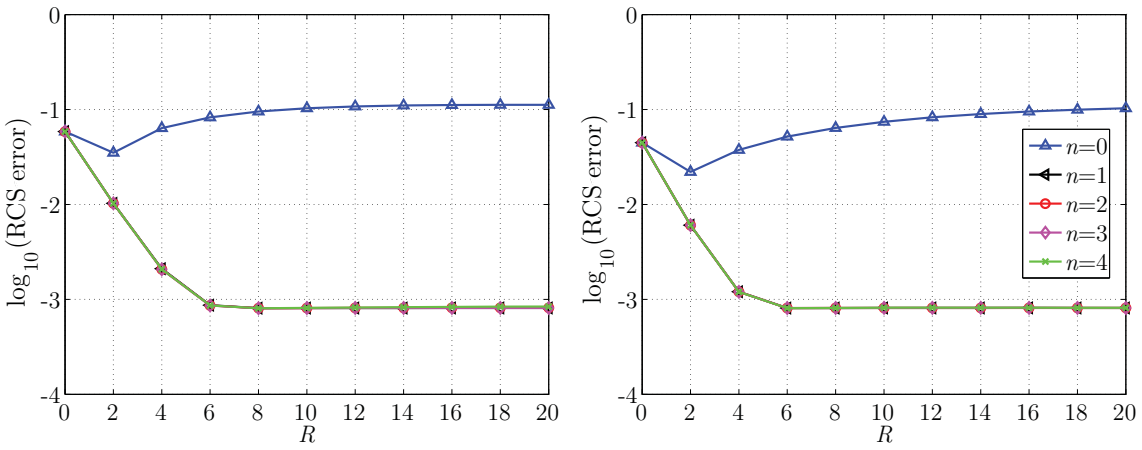
Figure E.10: Scattering by a PEC cylinder of diameter 8λ with optimal PML parameters

PML is increased to 2λ the error shows a weaker dependence on the PML parameters but the minimum error is not reduced in comparison with the computation with a thickness of λ.

Finally a cylinder of diameter 8λ is considered. Figure E.10 shows the computational mesh, transverse scattered field E_3, and the RCS. The solution is computed using a degree of interpolation $p = 6$ and the non-linear PML of thickness λ with optimized parameters $R = 10$ and $n = 2$. It is worth remarking that using the mesh of Figure E.10, a degree of interpolation $p = 6$ is sufficient to obtain a similar accuracy than by using $p = 8$ in the previous experiments, see also the comparison in the next section.

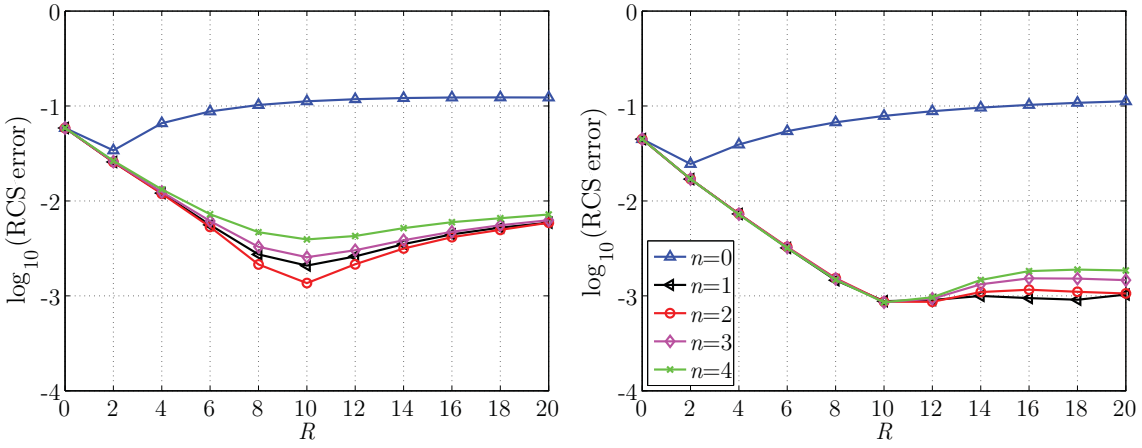
Figure E.11 shows the errors in the RCS, associated with the Berenger PML for a cylinder of diameter 8λ, as a function of the parameter R. The conclusions are similar to the previous experiments with cylinders of diameter 2λ and 4λ. An almost identical performance of the PML is obtained with a linear or quadratic degree of absorption and a value of $R \geq 6$ provides the minimum error, and consequently the best performance of the PML. Finally, if a PML of thickness 2λ is used no further reduction in error is achieved.

Figure E.12 shows the errors in the RCS, associated with the Berenger PML for a cylinder of diameter 8λ, as a function of the parameter R. The conclusions for the



(a) Thickness λ (b) Thickness 2λ

Figure E.11: Berenger PML: error in the RCS as a function of R and n for the cylinder of diameter 8λ, with a degree of interpolation $p = 6$



(a) Thickness λ (b) Thickness 2λ

Figure E.12: Non-linear PML: error in the RCS as a function of R and n for the cylinder of diameter 8λ, with a degree of interpolation $p = 6$

non-linear PML are also similar to those obtained with cylinders of diameter 2λ and 4λ. When a thickness of λ is considered, the maximum performance is achieved for a quadratic degree of absorption and a value of $R = 10$. By increasing the thickness of the PML to 2λ a weaker dependence of the RCS error on the PML parameters is

observed. A slightly better performance of the PML is obtained in comparison to the computations performed with a PML of thickness λ indicating that the error due to the PML dominates over the discretization error.

To conclude, for the Berenger PML, a linear or quadratic degree of absorption with $R = 6$ is an optimal choice, independent of thickness. In fact for $R \geq 6$ no difference in performance is obtained. It is worth recalling that several previous studies on the optimization of Berenger PML parameters can be found in the context of FD methods, see Taflove (1995) and references therein. In these studies, cubic or quartic degree of absorption was found to give optimal results for FD methods, contrasting with the results obtained in this section in the context of high-order DG methods. Also different values of the conductivity in the PML region were found to give best results. For FD methods the optimal value was found to be $R = 8$ or $R = 16$ depending on the thickness of the PML, whereas our numerical results indicate that for high-order DG methods $R \geq 6$ is optimal, independent of thicknesses. These results demonstrate that the optimal values of the PML parameters are highly dependent on the numerical methodology used to discretize Maxwell's equations.

For the non-linear PML no previous experience in the optimization of PML parameters is available. In this study, a quadratic degree of absorption with a value of $R = 8$ or $R = 10$ gives the best results. The use of a layer of thickness 2λ provides a more robust PML (i.e. with less dependence on the PML parameters).

E.4 Comparison

The numerical experiments in the previous section showed that, for a given spatial discretization, the Berenger PML is more precise than the non-linear PML. Nevertheless, it is important to remark that the Berenger PML requires more memory than the non-linear PML, for the same discretization. This is because the Berenger PML requires to store the nodal values of the extra non-physical variables in the absorbing layer. The number of degrees of freedom for the Berenger ($\mathtt{n}_{\mathtt{dof}}^{\mathrm{Ber}}$) and for the non-linear PML ($\mathtt{n}_{\mathtt{dof}}^{\mathrm{NL}}$) is given by

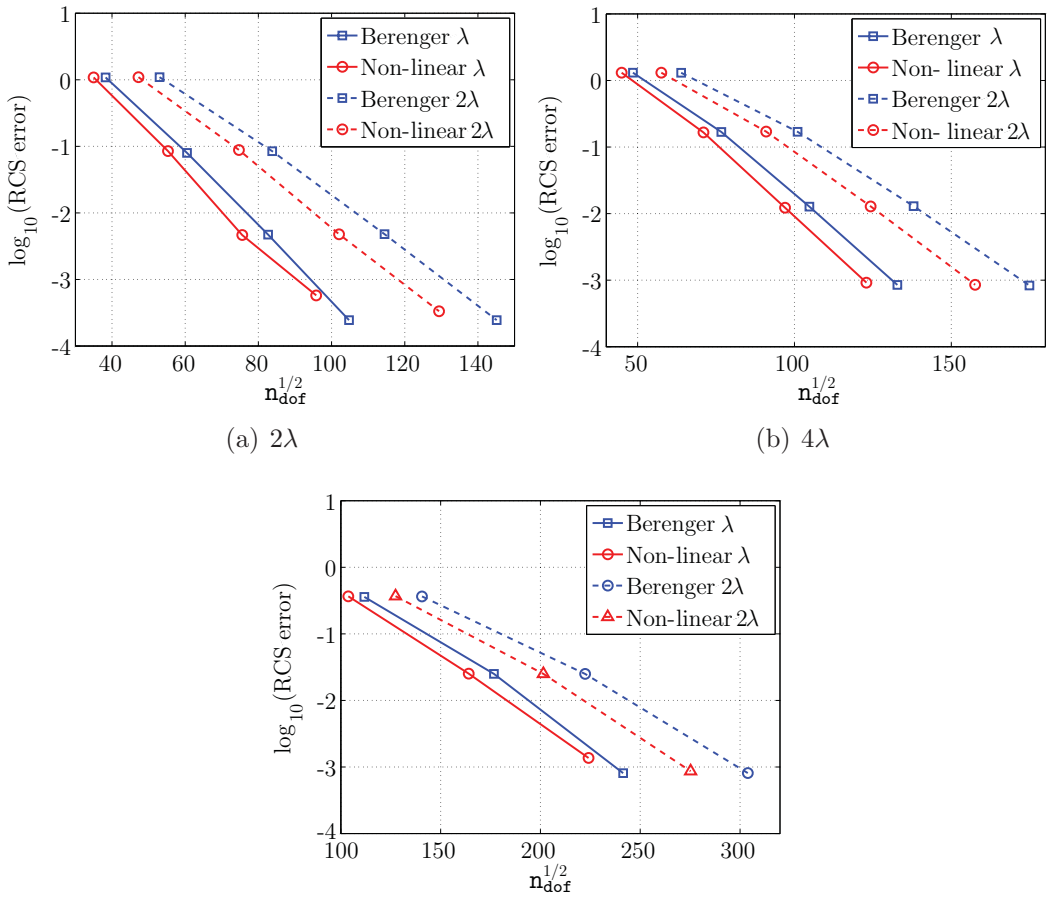
$$\mathtt{n}_{\mathtt{dof}}^{\mathrm{Ber}} = (\mathtt{n}_{\mathtt{el}}\mathtt{n}_{\mathtt{comp}} + \mathtt{n}_{\mathtt{el}}^{\mathrm{PML}}\mathtt{n}_{\mathtt{comp}}^{\mathrm{PML}})\mathtt{n}_{\mathtt{en}},$$

$$\mathtt{n}_{\mathtt{dof}}^{\mathrm{NL}} = \mathtt{n}_{\mathtt{el}}\mathtt{n}_{\mathtt{comp}}\mathtt{n}_{\mathtt{en}},$$

where $\mathtt{n}_{\mathtt{el}}$ is the total number of elements in the mesh, $\mathtt{n}_{\mathtt{comp}}$ is the number of (physical) components of the solution, $\mathtt{n}_{\mathtt{el}}^{\mathrm{PML}}$ is the number of elements in the PML region, $\mathtt{n}_{\mathtt{comp}}^{\mathrm{PML}}$ is the number of artificial variables introduced in the Berenger PML.

In 2D $\mathtt{n}_{\mathtt{comp}} = 3$ and $\mathtt{n}_{\mathtt{comp}}^{\mathrm{PML}} = 1$. Therefore, for triangular meshes, the Berenger PML introduces $(p+1)(p+2)/2$ extra degrees of freedom for each element in the PML region. In 3D the situation is worse since the number of extra unknowns introduced in the Berenger PML is exactly the same as the number of physical variables ($\mathtt{n}_{\mathtt{comp}} = 6$, $\mathtt{n}_{\mathtt{comp}}^{\mathrm{PML}} = 6$). Thus, for tetrahedral meshes, the Berenger PML introduces $(p+1)(p+2)(p+3)$ extra degrees of freedom for each element in the absorbing layer. In such cases, the memory requirements of a Berenger PML of thickness λ are approximately the same as for a non-linear PML of thickness 2λ. Furthermore, in a DG context, the Berenger PML is slightly more computationally expensive than the non-linear PML, due to the computation of extra numerical fluxes, associated to the non-physical variables introduced in the PML region, see Appendix E.1.1.

A p-convergence comparison is next performed using the optimized parameters for the Berenger and the non-linear PMLs with thicknesses λ and 2λ. First the cylinder of diameter 2λ is considered. For each degree of interpolation the optimal parameters for both PMLs are extracted from the numerical experiments shown in Figures E.5 and E.6. The results shown in Figure E.13 (a) indicate the expected exponential convergence behavior for a problem with a smooth solution, see Szabó and Babuška (1991), when both PMLs are used. However, for the non-linear PML with thickness of λ, the exponential convergence deteriorates when the polynomial degree is increased from $p = 6$ to $p = 8$. This indicates that, for this thickness, the error associated with the PML dominates over the discretization error. Secondly, a p-convergence study is performed for the cylinder of diameter 4λ. Again, results shown in Figure E.13 (b) indicate the expected (exponential) convergence when the optimized parameters



(a) 2λ

(b) 4λ

(c) 8λ

Figure E.13: p-convergence comparison for a serie of PEC cylinders: error in the RCS as a function of the number of degrees of freedom (increasing p starting with 2 and in increments of 2) using the Berenger and the non-linear PML with different thicknesses

from FiguresE.8 and E.9 are used. As in the case of the 2λ cylinder, the non-linear PML outperforms the Berenger PML. Finally, a p-convergence study is performed for the cylinder of diameter 8λ using the optimized parameters from Figures E.11 and E.12. The results of this investigation are shown in Figure E.13 (c), which once again indicates that the non-linear PML is clearly the better choice. Nevertheless, it is worth noting that the exponential convergence deteriorates slightly when the polynomial degree is increased from $p = 4$ to $p = 6$.

When using the optimum PML parameters and performing p refinement, the

achievable relative error for the RCS was found to be at least 10^{-3}. Further p refinement may also lead to a further reductions in the error, but this level of accuracy has been previously reported as being sufficient for engineering purposes (Hesthaven and Warburton, 2002). Furthermore this level of accuracy is better than achieved with other numerical schemes, see Hachemi et al. (2004) and Woo et al. (1993).

Adopting this as the required level of accuracy, then it is clear that the non-linear PML of thickness λ is the best choice for the two dimensional test case considered here, provided that the optimized parameters are adopted. If optimized parameters are not available, the Berenger PML of thickness of λ becomes a better choice as it is robust across a wider range of parameters. The 2λ thick non-linear PML is even more robust, although, in 2D it is more expensive than the λ thick Berenger PML. In 3D, a λ thick Berenger PML has similar storage requirements as a 2λ thick non-linear PML, making the latter a computational viable alternative, even if optimized parameters are not available.

Appendix F

Discontinuous Galerkin NEFEM for compressible flow

An accurate description of the geometrical model is crucial in some applications, such as the numerical solution of Euler equations of gas dynamics. In a DG framework, using a linear approximation for the geometry it is not possible to converge to the steady state solution, even if the mesh is drastically refined near the curved boundary (Bassi and Rebay, 1997). Van der Ven and Van der Vegt (2002) present a detailed study of this problem to conclude that accurate results can only be obtained taking into account the curvature of the domain. More recently, Krivodonova and Berger (2006) propose a new methodology for the computation of the fluxes along curved boundaries but, unfortunately, the proposed method is not conservative and it is only well suited for inviscid computations.

The importance of the geometrical model in the numerical solution of compressible Euler equations is not exclusive of DG methods. Barth (1998) identify the problem in the context of Finite Volume (FV) methods, and more recent advances in this area can be found in Wang and Liu (2006).

Here NEFEM is proposed as an efficient alternative for a proper treatment of curved wall boundaries in compressible flow problems using a DG formulation. First, Euler equations and their DG formulation are briefly recalled, highlighting the differences with respect to the DG formulation of Maxwell's equations presented in Appendix D. The implementation of boundary conditions is discussed for both subsonic

and supersonic flows, in the case of inflow, outflow and wall boundaries. Finally several numerical examples are considered. Standard test cases are used to validate the developed DG NEFEM code for the solution of Euler equations, including solutions with shock discontinuities. A comparison between isoparametric FEs and NEFEM is presented, complementing the comparison presented by Sevilla et al. (2008b). Further examples are included in order to show the possibilities of NEFEM for compressible flow applications.

F.1 Euler equations

Euler equations of gas dynamics express the conservation of mass, momentum and energy in a compressible, inviscid and non-conducting fluid. In the absence of external volume forces, the strong form of these conservation laws is

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}) = 0 \\ &\frac{\partial \rho \boldsymbol{v}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v} \otimes \boldsymbol{v} + \mathrm{p}\boldsymbol{I}) = 0 \\ &\frac{\partial \rho E}{\partial t} + \boldsymbol{\nabla} \cdot \big((\rho E + \mathrm{p})\boldsymbol{v}\big) = 0 \end{aligned} \tag{F.1}$$

where ρ is the density, $\rho\boldsymbol{v}$ is the momentum, ρE is the total energy per unit volume and p is the pressure, see Donea and Huerta (2005) for more details. The system of nonlinear hyperbolic equations (F.1) can be rewritten as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_k(\boldsymbol{U})}{\partial x_k} = \boldsymbol{0}, \tag{F.2}$$

where $\boldsymbol{U}$ is the vector of conservation variables and $\boldsymbol{F}_k$ are the associated flux vectors for each spacial dimension. They are defined as follows:

$$\boldsymbol{U} = \begin{pmatrix} \rho \\ \rho \boldsymbol{v} \\ \rho E \end{pmatrix}, \quad \boldsymbol{F}_k(\boldsymbol{U}) = \begin{pmatrix} \rho v_k \\ \rho \boldsymbol{v} v_k + \boldsymbol{e}_k \mathrm{p} \\ (\rho E + \mathrm{p}) v_k \end{pmatrix},$$

where $\boldsymbol{e}_k$ is the unitary vector in the x_k direction.

An equation of state, relating the internal energy to pressure and density, completes this system of nonlinear hyperbolic equations. For a perfect polytropic gas the equation of state is

$$\mathrm{p} = (\gamma - 1)\rho\Big(E - \frac{1}{2}\|\boldsymbol{v}\|^2\Big),$$

where γ is the ratio of the specific heat coefficients (specific heat at constant pressure over specific heat at constant volume), with value $\gamma = 1.4$ for air.

A usual quantity for postprocess of flow computations is the Mach number, defined as

$$M = \frac{\|\boldsymbol{v}\|}{\mathrm{c}},$$

where $\mathrm{c} = \sqrt{\gamma \mathrm{p}/\rho}$ is the speed of sound. Other useful quantity for the evaluation of the accuracy of subsonic flow computations is the entropy error

$$\epsilon_{ent} = \frac{\mathrm{p}}{\mathrm{p}_\infty}\left(\frac{\rho_\infty}{\rho}\right)^\gamma - 1, \tag{F.3}$$

where the superscript ∞ indicates free-stream values, see Van der Ven and Van der Vegt (2002).

For a detailed presentation of the Euler equations see classical text books such as Anderson (1982), Hirsch (1988) and Laney (1998). For a presentation of the Euler equations in a FE context, see also Donea and Huerta (2005).

F.2 Discontinuous Galerkin formulation

The DG weak formulation of a general system of conservation laws (F.2) is deduced in Section D.2. It can be written as

$$\int_{\Omega_e} \boldsymbol{W} \cdot \frac{\partial \boldsymbol{U}_e}{\partial t}\, d\Omega - \int_{\Omega_e} \frac{\partial \boldsymbol{W}}{\partial x_k} \cdot \boldsymbol{F}_k(\boldsymbol{U}_e)\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{W} \cdot \widetilde{\boldsymbol{F}}_n(\boldsymbol{U}_e, \boldsymbol{U}_e^{\text{out}})\, d\Gamma = 0, \quad \text{(F.4)}$$

where $\boldsymbol{U}_e$ denotes the restriction of $\boldsymbol{U}$ to the element Ω_e, $\boldsymbol{U}_e^{\text{out}}$ denotes the restriction of $\boldsymbol{U}$ to neighboring elements, see Equation (D.16) in Section D.2, and $\widetilde{\boldsymbol{F}}_n(\boldsymbol{U}_e, \boldsymbol{U}_e^{\text{out}})$ is the numerical flux, see Appendix D for the details. Contrary to Maxwell's equations, the system of Euler equations is non-linear, and therefore, the definition of a proper numerical flux is a crucial issue to get accurate results. Some of the more popular flux functions used in the literature for the Euler equations are the exact Riemann solver, the Roe solver, the Lax-Friederichs solver or the Harten-Lax-van Leer (HLLE) solver. For a complete description of several numerical flux functions for non-linear hyperbolic problems see Toro (1997).

The DG formulation (F.4) is discretized in each element, leading to a system of ODEs

$$\mathbf{M}\frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \quad \text{(F.5)}$$

where $\mathbf{U}$ is the vector of nodal values (or approximation coefficients in a more general case), $\mathbf{M}$ is a block diagonal mass matrix and $\mathbf{R}(\mathbf{U})$ is the residual vector. The ODE system (F.5) is advanced in time using the explicit third-order total variation diminishing Runge-Kutta (TVD-RK) scheme presented by Cockburn and Shu (1989). Nevertheless, it is worth noting that semi-implicit and implicit time integration schemes seems to be an efficient alternative for steady state computations, see Dolejší and Feistauer (2004) and Persson and Peraire (2006a). The stability condition for the TVD-RK method requires a time step

$$\Delta t \leq \frac{h}{p(2p+1)(v_n + \mathrm{c})}, \quad \text{(F.6)}$$

where h is the minimum element size, and p is the degree of the polynomial approximation.

F.2.1 Boundary conditions

This section recalls the implementation of boundary conditions for the Euler equations. As usual in the solution of hyperbolic equations, a characteristic analysis is performed at the boundary to decide the quantities to be prescribed, see for instance Hirsch (1988) or Donea and Huerta (2005). Euler equations are first diagonalized in the normal direction to the boundary, and three distinct eigenvalues are obtained

$$\lambda_1 = v_n - \mathrm{c}, \qquad \lambda_2 = \lambda_3 = \lambda_4 = v_n, \qquad \lambda_5 = v_n + \mathrm{c},$$

where $v_n = \boldsymbol{v} \cdot \boldsymbol{n}$. Then, at each point on the boundary the number of prescribed boundary conditions corresponds to the number of negative eigenvalues.

On a supersonic inflow boundary, that is $v_n < 0$ and $|v_n| > \mathrm{c}$, all eigenvalues are negative. Therefore, all the conserved quantities are imposed to free-stream values. On the contrary, on a supersonic outflow boundary all eigenvalues are positive, and therefore, no boundary conditions are required. The situation is more difficult for subsonic flows, $|v_n| < \mathrm{c}$. At an inflow boundary only the eigenvalue λ_1 is positive, and four boundary conditions must be prescribed, corresponding to the associated Riemmann variables. Finally, at an outflow boundary only the eigenvalue λ_1 is negative and therefore, its associated Riemmann variable is imposed at the boundary. Finally, on a wall boundary the condition is that velocity is tangential to the boundary, that is $v_n = 0$. Only the first eigenvalue λ_1 is negative, and therefore, only one boundary condition must be prescribed.

From an implementation point of view the ideas in LeVeque (2002), initially developed in the context of FV methods, are considered. Fictitious elements are considered along the boundary, and the value of the solution is set to impose the boundary conditions through the numerical flux. For a supersonic inflow boundary the external

state is defined using the free-stream values, that is $\boldsymbol{U}^{\text{out}} = \boldsymbol{U}^{\infty}$. On a supersonic supersonic boundary the external state is set to the interior state, $\boldsymbol{U}^{\text{out}} = \boldsymbol{U}_e$. For a subsonic inflow boundary, the Riemmann variables associated to negative eigenvalues are computed from free-stream conditions and the Riemmann variables associated to positive eigenvalues are computed from the internal state. Thus, on a subsonic inflow boundary the exterior state, $\boldsymbol{U}^{\text{out}} = (\rho^{\text{out}}, \rho^{\text{out}}\boldsymbol{v}^{\text{out}}, \rho^{\text{out}}E^{\text{out}})^T$, is computed as

$$\rho^{\text{out}} = \frac{(\rho^{\infty})^{\gamma}(\gamma-1)^2}{16\gamma \text{p}^{\infty}}(w_1^e - w_5^{\infty})^2,$$

$$\boldsymbol{v}^{\text{out}} = \boldsymbol{v}_t^{\infty} + \frac{1}{2}(w_1^e + w_5^{\infty})\boldsymbol{n},$$

$$E^{\text{out}} = \frac{\gamma-1}{16\gamma}(w_1^e - w_5^{\infty})^2 + \frac{1}{2}\|\boldsymbol{v}\|^2,$$

where

$$w_1^e = \boldsymbol{n}\cdot\boldsymbol{v}^e + \frac{2\text{c}^e}{\gamma-1}, \qquad w_5^{\infty} = \boldsymbol{n}\cdot\boldsymbol{v}^{\infty} - \frac{2\text{c}^{\infty}}{\gamma-1}.$$

Similarly, for a subsonic outflow boundary the external state can be computed as

$$\rho^{\text{out}} = \rho^e\left(\frac{\text{p}^{\infty}}{\text{p}^e}\right)^{1/\gamma},$$

$$\boldsymbol{v}^{\text{out}} = \boldsymbol{v}^e + \frac{2}{\gamma-1}\left(\text{c}^e - \sqrt{\frac{\gamma\text{p}^{\infty}}{\rho^{\infty}}}\right)\boldsymbol{n},$$

$$E^{\text{out}} = \frac{\text{p}^{\infty}}{\rho^{\text{out}}(\gamma-1)} + \frac{1}{2}\|\boldsymbol{v}^{\text{out}}\|^2.$$

Finally, at a wall boundary the external state is computed as

$$\rho^{\text{out}} = \left(\frac{(\rho^e)^{\gamma}}{\gamma\text{p}^e}(\text{c}^{\text{out}})^2\right)^{1/(\gamma-1)},$$

$$\boldsymbol{v}^{\text{out}} = \boldsymbol{v}^e - (\boldsymbol{n}\cdot\boldsymbol{v}^e)\boldsymbol{n},$$

$$E^{\text{out}} = \frac{(\mathrm{c}^{\text{out}})^2}{\gamma(\gamma-1)} + \frac{1}{2}\|\boldsymbol{v}^{\text{out}}\|^2.$$

where

$$\mathrm{c}^{\text{out}} = \mathrm{c}^e + \frac{\gamma-1}{2}\boldsymbol{n}\cdot\boldsymbol{v}^e.$$

F.3 Implementation details

In this section the computation of the residual vector of the DG semi-discrete system (F.5) is detailed. Main differences with respect to the computation of the residual for Maxwell's equations are described, see Section F.3.1. Finally, the flowchart of the developed DG code for solving Euler equations is presented, and the differences with the flowchart of the developed code for solving Maxwell's equations are discussed.

F.3.1 Residual computation

The evaluation of the residual $\mathbf{R}(\mathbf{U})$ in Equation (F.5), which involves the flux at the interior of the elements and their boundaries, can be carried out, as usual, with two non-equivalent options: a quadrature-free implementation or a full quadrature version, see Section D.3.1 and Donea and Huerta (2005). With a quadrature-free implementation the flux at the integration points is interpolated in terms of the flux at nodal values, whereas with a full quadrature version fluxes are evaluated at integration points, in terms of the solution at each integration point. The use of a quadrature-free implementation leads to an important save in computational cost thanks to the use of elemental matrices, instead of a loop on integration points. Moreover, for triangles with straight sides (or tetrahedrons with planar faces) these elemental matrices can be computed, using the Jacobian, from matrices previously computed at the reference element, see Atkins and Shu (1998), with an important reduction in computational time. However, numerical experiments reveal that a quadrature-free implementation for the Euler equations suffers from instability problems in the vicinity of stagnation points, see Hillewaert et al. (2006) for a detailed explanation. Thus, all computations

presented here are obtained with a full quadrature implementation, see more details in Section D.3.1.

Remark 5. *It is well known that the standard nodal approximation presents oscillations when it is used to approximate the solution in the vicinity of a shock (discontinuity). In the numerical examples shown in this appendix, an artificial viscosity is introduced in order to avoid the problem. The viscous term is discretized using the local DG (LDG) method proposed by Cockburn and Shu (1998), although the interior penalty method (IPM) (Arnold, 1982), or the compact DG (CDG) method proposed by Peraire and Persson (2008) method, are also efficient approaches to treat the viscous terms in a DG formulation, see Montlaur et al. (2008) and Montlaur (2009) for further details. Shock sensors and efficient procedures are also desired in this context, see Persson and Peraire (2006b), Casoni et al. (2009) and references therein.*

F.3.2 Developing a code for solving Euler equations

The flow chart of the developed code for solving Euler equations with a DG formulation is depicted in Figure F.1.

Note that the flowchart for the developed codes for solving Maxwell and Euler equations are very similar. In fact, both systems of equations can be written as a hyperbolic system of conservation laws. The most important difference between Maxwell and Euler equations is that Maxwell equations are linear and Euler equations are non-linear due to the non-linearity of the momentum equation, see Section F.1. The first consequence is that a non-constant time step is needed to advance de solution in time with explicit time marching algorithms, see Equation (F.6).

The most relevance consequence of the non-linearity of Euler equations is that a full quadrature implementation must be adoped, see Section F.3.1 and Hillewaert et al. (2006). Thus, although the flow chart for the Euler code is simpler than the flow chart of the Maxwell code (see Figure D.3), the loops on faces and elements are more

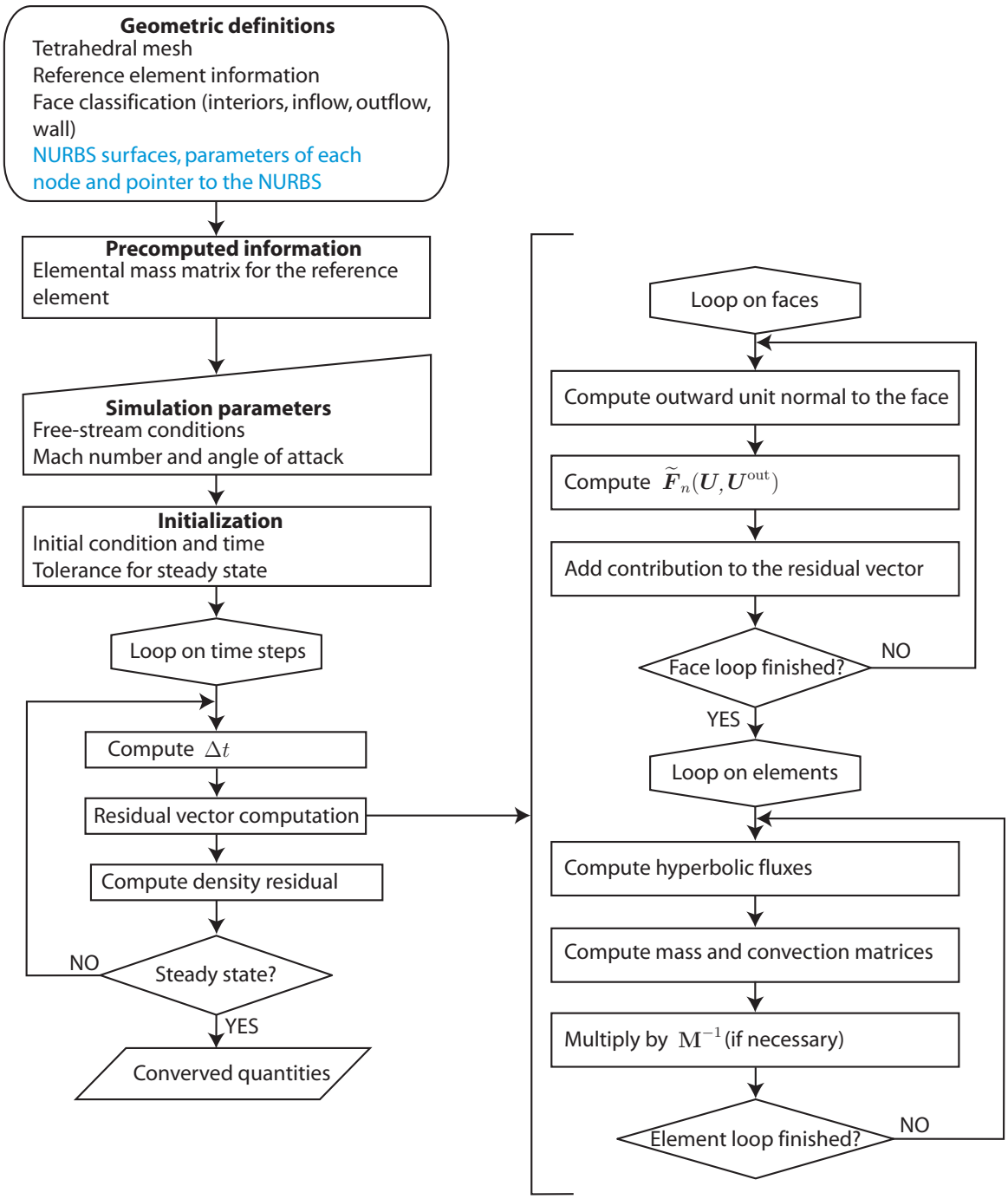


Figure F.1: Flowchart of the developed code for solving Euler equations with a DG formulation

costly due to the full quadrature implementation. Recall that for Maxwell's equations the full quadrature implementation is only introduced in curved elements. Thus, the loops of the Euler code need to perform numerical integration even for elements with planar faces. Therefore, the precomputation in the Euler code is restricted to the mass matrix of the reference element to use in the computation of $\mathbf{M}^{-1}$ for planar elements.

F.4 Numerical examples

This section presents several examples that show the performance of NEFEM for the solution of compressible flow problems. First, some classical test cases are considered to validate the developed DG code. Secondly, the application of NEFEM to Euler equations is considered, and a comparison of NEFEM and isoparametric FEs is presented. Finally, further examples involving more complex problems are shown.

F.4.1 Validation

Several test cases from NPARC Alliance (2008) are considered to test the performance of the DG code. The first example is the classical shock tube problem that offers an interesting flow structure, which is characterized by three steadily moving waves: a shock wave, a contact discontinuity, and an expansion fan, see Laney (1998). Based on the instantaneous location of each wave, it is possible to divide the general flow within a shock tube into five sections, which are illustrated in Figure F.2. An analytical solution to the problem exists up to the point in time where either the shock wave or the head of the expansion is reflected by the respective end of the tube, see Anderson (1982).

The computational domain is $\Omega = [0,1]\times[0,0.4]$ and the initial condition is defined as

$$\boldsymbol{U}(\boldsymbol{x},0) = \begin{cases} \big(3,0,0,3/(\gamma-1)\big)^T & \text{if} \quad x \leq 0.5 \\ \big(1,0,0,1/(\gamma-1)\big)^T & \text{if} \quad x > 0.5, \end{cases}$$

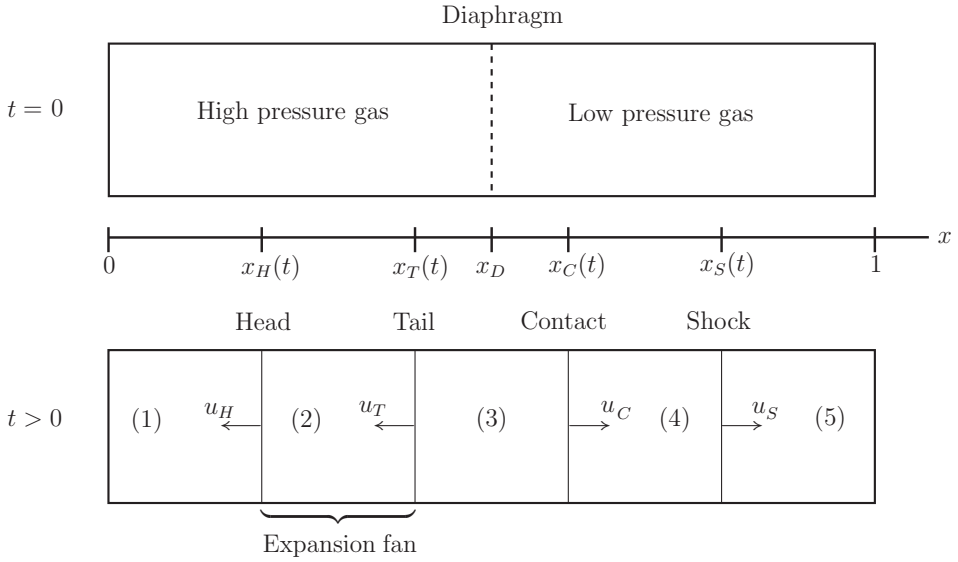
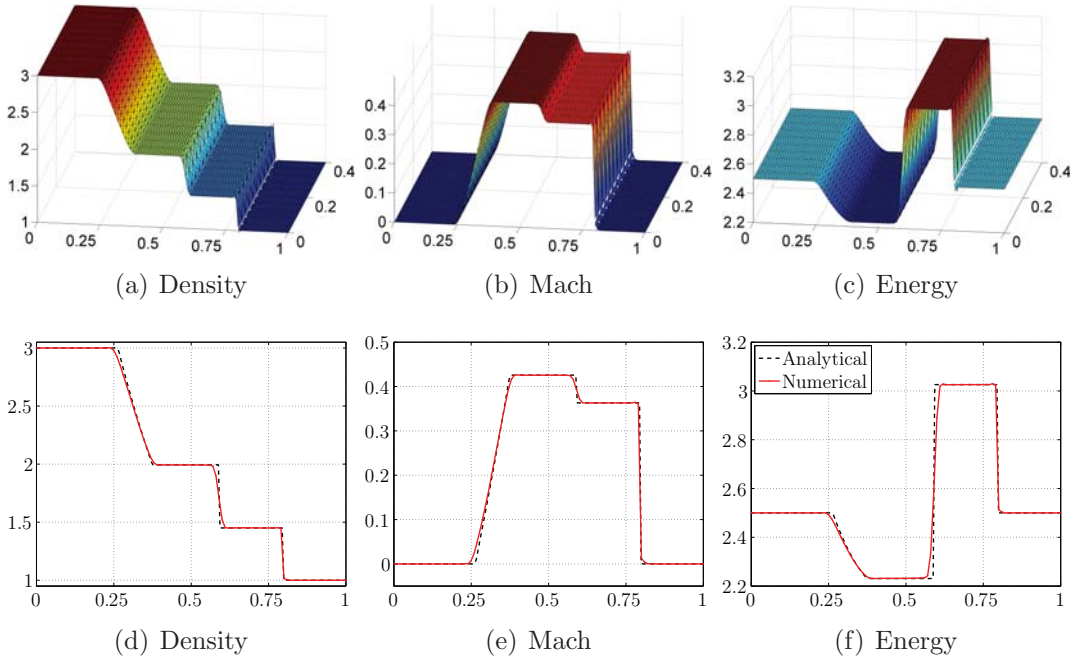


Figure F.2: Shock tube problem setup

see LeVeque (1992).

Figure F.3 shows the numerical solution when the full flow structure is developed, at time $t = 0.2$. The mesh size is $h = 0.01$ and the solution is computed with linear approximation. The time step Δt is selected as the stability limit defined in Equation (F.6). Density, Mach number and energy are represented for linear approximation, and compared with the analytical solution along the mid-section of the domain. Good agreement is observed between analytical and numerical solutions. Owing to the presence of discontinuities, a dissipation mechanism is needed in order to avoid the oscillations when the classical nodal interpolation is applied. In this example an artificial viscosity is added and the LDG formulation is considered, see Remark 5 in Section F.2.

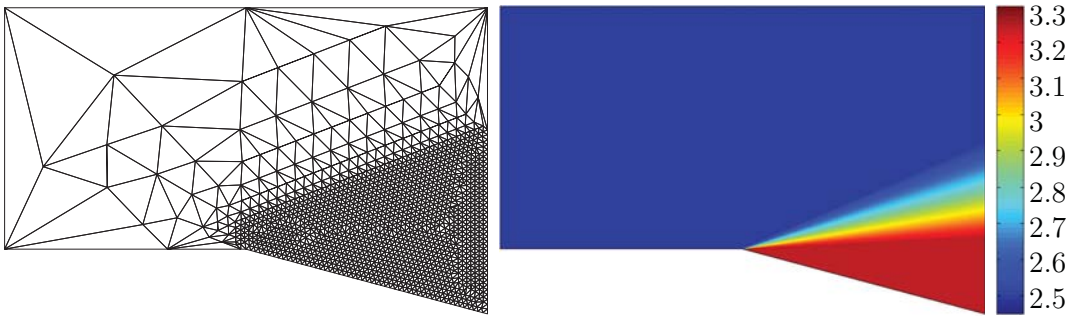
The second test case is the so-called Prandtl-Meyer 15 degree expansion, consisting on a supersonic flow at Mach 2.5 that forms an expansion fan. The setup of the problem and analytical values for comparison are extracted from the archive of NPARC Alliance (2008) and Anderson (1982). For instance, the analytical value of the maximum Mach number is 3.24. Figure F.4 shows the computational mesh and the Mach number distribution of the steady state numerical solution for a degree of



(a) Density (b) Mach (c) Energy

(d) Density (e) Mach (f) Energy

Figure F.3: Shock tube problem: numerical solution (top) and comparison of the mid-section with the analytical solution (bottom)

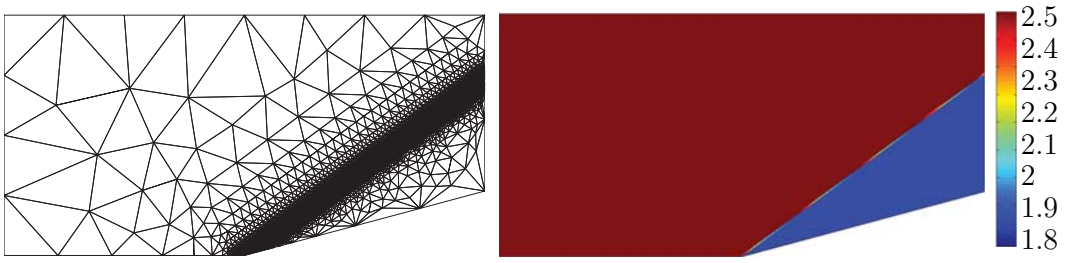


(a) Mesh (b) Mach number

Figure F.4: Prandtl-Meyer 15 degree corner expansion at Mach 2.5

approximation $p = 6$. The computed value of the Mach number shows a maximum of 3.23, in excellent agreement with the analytical solution.

As a third validation example the oblique shock caused by a 15 degree compression corner is considered, also extracted from the NPARC database (NPARC Alliance, 2008). Again, there is an analytical solution for this problem (Anderson, 1982). For

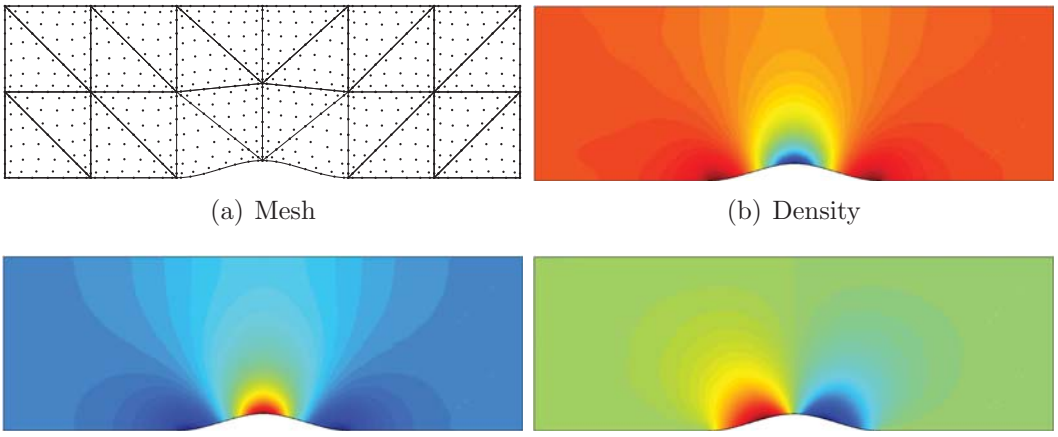


(a) Mesh (b) Mach number

Figure F.5: Oblique shock on a 15 degree wedge at Mach 2.5

instance, the angle of the oblique shock is 36.94 degrees and the Mach number behind the shock is 1.87. Figure F.5 shows the computational mesh and the Mach number distribution of the steady state numerical solution for linear approximation. Note that, in this example, the mesh is drastically refined towards the shock. Moreover, the degree of the approximation is reduced to $p = 0$ near the discontinuity, in order to properly capture the shock without spurious oscillations. The predicted Mach number is 1.87 and the angle of the shock is 36.91, showing, again, an excellent agreement with the analytical values.

The final test considers the subsonic flow over a sinusoidal bump at free stream Mach number $M^{\infty} = 0.3$. NEFEM exactly deals with the curved boundary, corresponding to a sinusoidal bump. Figure F.6 (a) shows the computational mesh, with only two curved elements and Fekette nodal distributions. The numerical solution is also represented, namely, the density, the Mach number and the horizontal velocity, for a NEFEM solution with $p = 9$. Since the flow is subsonic and inviscid, a measure of the quality of the solution is the entropy error, defined in Equation (F.3). Figure F.7 shows the p-convergence in entropy error as p is uniformly increased starting with $p = 2$ up to $p = 9$, in the discretization shown in Figure F.6 (a). As expected for a problem with a smooth solution, optimal (exponential) convergence is achieved.



(a) Mesh (b) Density

(c) Mach number (d) Horizontal velocity

Figure F.6: Subsonic flow over a sinusoidal bump: NEFEM solution with $p = 9$

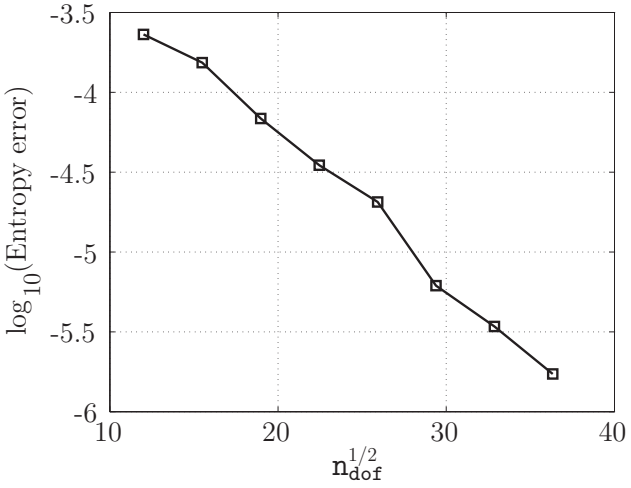
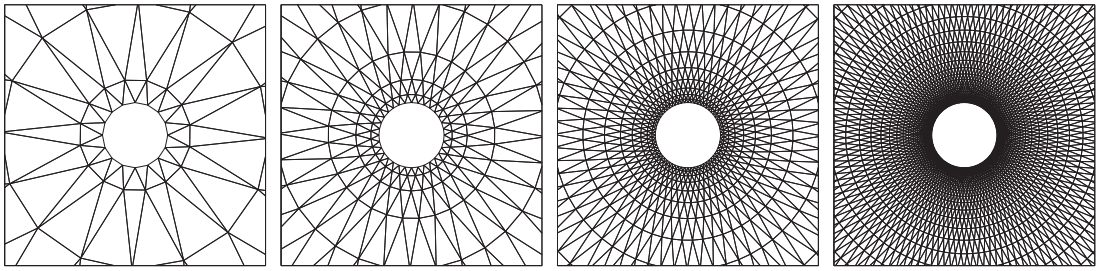


Figure F.7: Subsonic flow over a sinusoidal bump: NEFEM p-convergence of the entropy error

F.4.2 NEFEM performance comparison

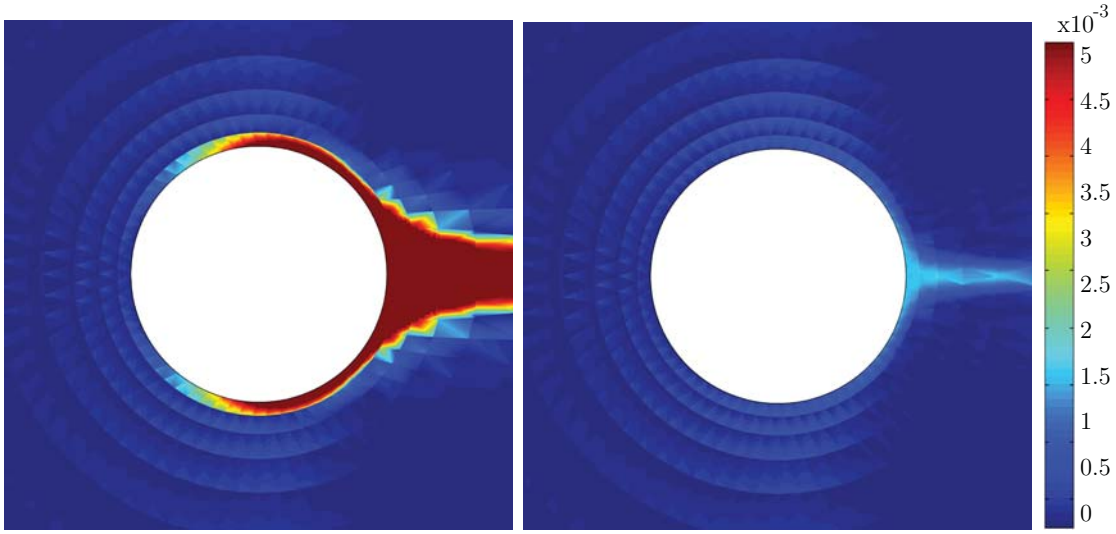
This section presents a comparison of isoparametric FEs and NEFEM for the numerical solution of a classical test problem: the subsonic flow around a circle at free-stream Mach number $M^\infty = 0.3$.

Here a linear approximation of the solution is considered and the influence of the numerical flux is investigated for both isoparametric FEs and NEFEM. Two classical



(a) Mesh 1 (b) Mesh 2 (c) Mesh 3 (d) Mesh 4

Figure F.8: Inviscid subsonic flow around a circle: detail of four meshes for the h-refinement study

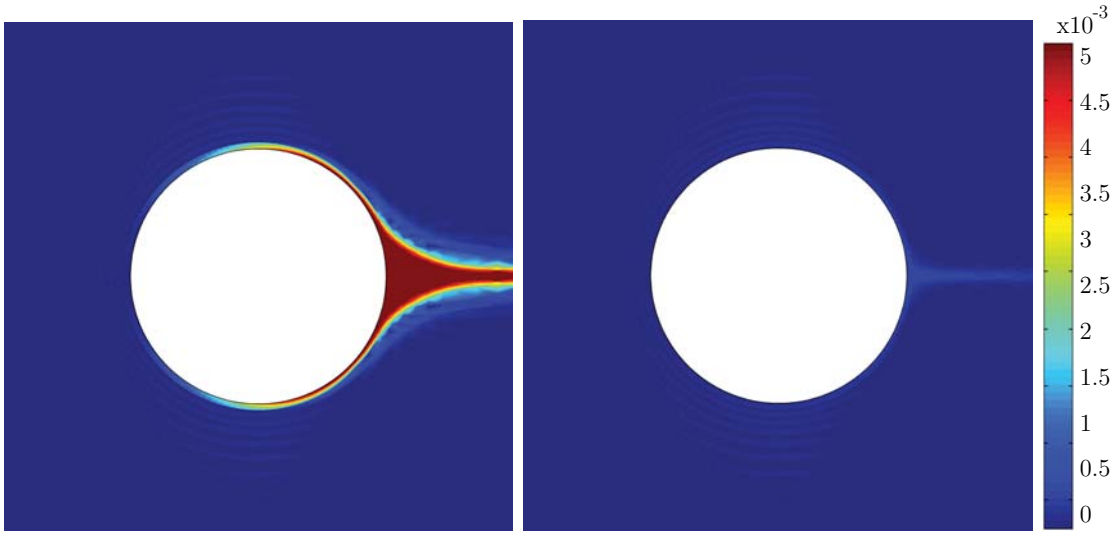


(a) FEM, Mesh 3 (b) NEFEM, Mesh 3

Figure F.9: Inviscid subsonic flow around a circle: detail of the entropy error distribution near the circle, for FEM and NEFEM in the mesh shown in F.8 (c)

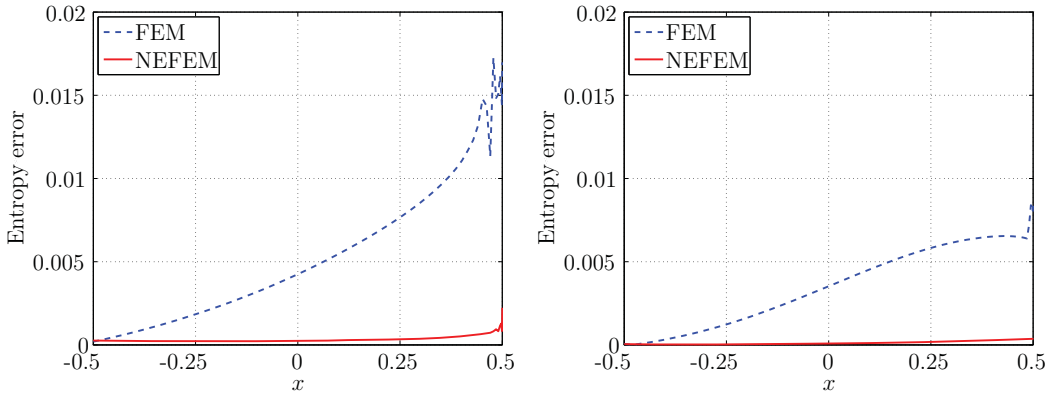
Riemmann flux functions are considered, namely the Lax-Friederichs flux and the Roe flux (Toro, 1997). Four meshes with 16×4, 32×8, 64×16, and 128×32 nodes (i.e. 128, 512, 2048 and 8192 elements respectively) are considered, see Figure F.8.

Figures F.9 and F.10 show a detailed view of the entropy error distribution near the circle, for isoparametric FEs and NEFEM in the discretizations shown in Figures F.8 (c) and (d) and by using the Roe flux. In NEFEM, the exact computation of the outward unit normal improves the imposition of the solid wall boundary condi-



(a) FEM, Mesh 4 (b) NEFEM, Mesh 4

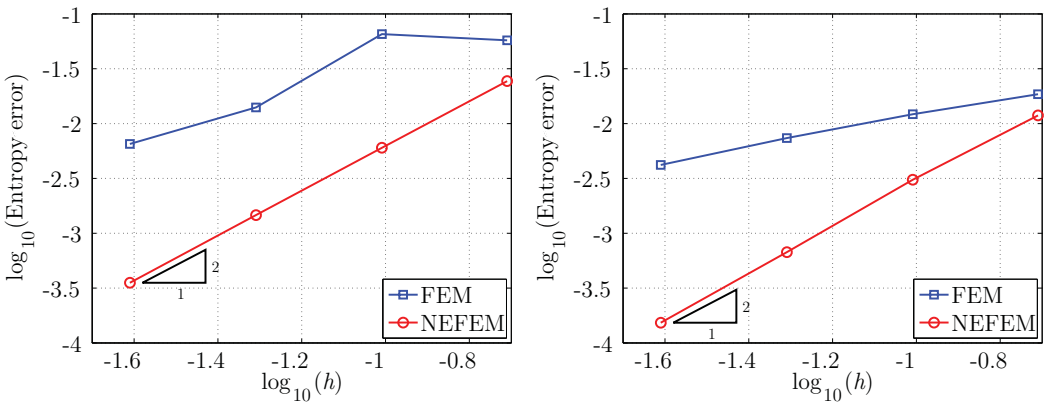
Figure F.10: Inviscid subsonic flow around a circle: detail of the entropy error distribution near the circle, for FEM and NEFEM in the mesh shown in F.8 (d)



(a) Lax-Friederichs (b) Roe

Figure F.11: Inviscid subsonic flow around a circle: comparison of the entropy error for isoparametric FEs and NEFEM using different numerical fluxes in the mesh shown in Figure F.8 (d)

tion. The exact boundary representation drastically reduce the entropy production compared to isoparametric FEs. Figure F.11 shows a comparison of the entropy error on the upper part of the circle, for isoparametric FEs and NEFEM in the fine mesh shown in Figure F.8 (d) using different numerical fluxes. Note that an important

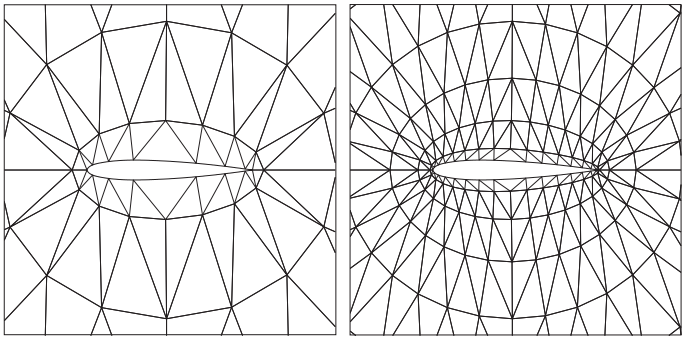


(a) Lax-Friederichs (b) Roe

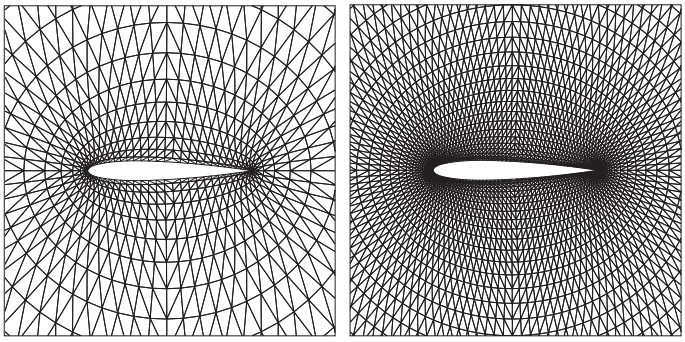
Figure F.12: Inviscid subsonic flow around a circle: h-convergence of the entropy error for isoparametric FEs and NEFEM using different numerical fluxes

reduction of the entropy error is observed by using the Roe flux compared to the Lax-Friederichs flux. In fact, the results obtained with isoparametric FEs in the meshes shown in Figures F.8 (a) and (b) exhibit a strong dependence on the numerical flux function. But it is worth remarking that NEFEM is more than one order of magnitude more accurate than the corresponding isoparametric FEs in the discretizations shown in Figures F.8 (c) and (d), with no dependence on the numerical flux function.

Figure F.12 shows the evolution of the entropy error on the upper part of the circle as a function of the mesh size. As observed by Bassi and Rebay (1997), the optimal rate of convergence is not obtained with linear isoparametric FEs. Entropy production observable in Figures F.9 (a) and F.10 (a) deteriorates the h-convergence of isoparametric FEs. In contrast, NEFEM exhibits the optimal convergence rate for linear approximation. Sevilla et al. (2008b) present a similar analysis in terms of other aerodynamic quantities, such as the pressure loss and the pressure coefficient, and similar conclusions are derived.



(a) Mesh 1 (b) Mesh 2



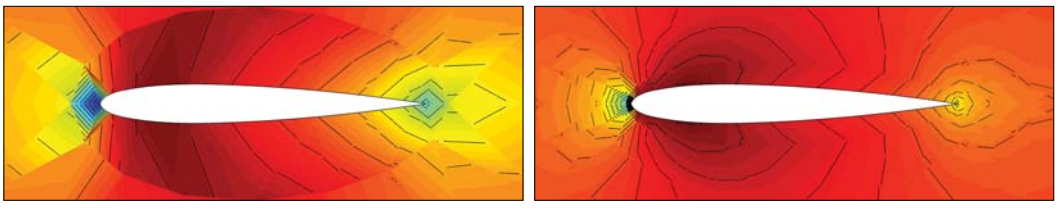
(c) Mesh 3 (d) Mesh 4

Figure F.13: Inviscid subsonic flow around a NACA0012 airfoil: detail of four meshes for the h-refinement study

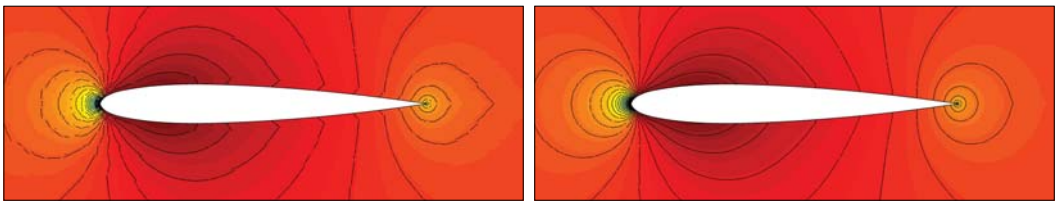
F.4.3 Further examples

This section presents the application of NEFEM to more complex applications, involving the subsonic flow around airfoils.

The flow around a NACA0012 airfoil at $M^\infty = 0.3$ is considered first. The discretizations shown in Figure F.13 are used in the computations. To design such meshes, a conformal mapping (Benson, 2008). The Mach number distribution and isolines for a NEFEM solution with linear approximation is represented in Figure F.14, illustrating the h convergence as the mesh is refined. Entropy errors are represented in Figure F.15 for FEM and NEFEM solutions with $p = 1$ in the discretizations shown in Figures F.13 (c) and (d). Again, the linear approximation of the geometry with isoparametric elements produces entropy that prevents convergence to the

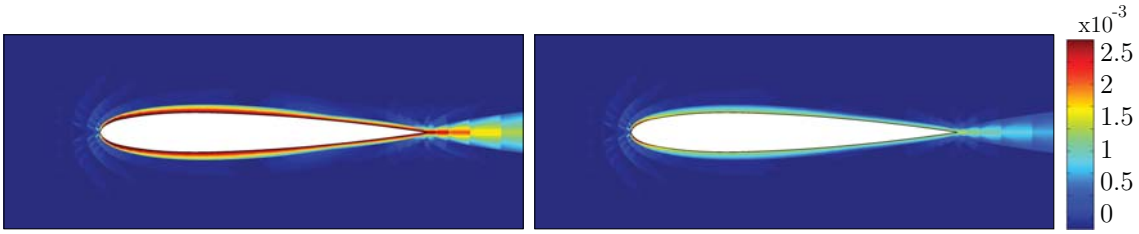


(a) Mesh 1 (b) Mesh 2

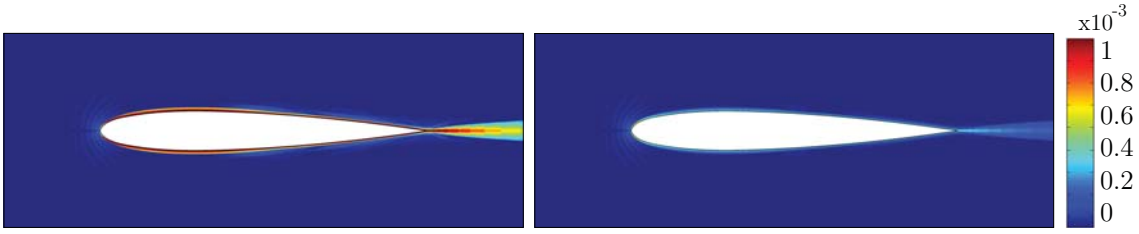


(c) Mesh 3 (d) Mesh 4

Figure F.14: Inviscid subsonic flow around a NACA0012 airfoil: detail of the Mach number distribution and isolines for NEFEM with linear approximation



(a) Mesh 3, FEM (b) Mesh 3, NEFEM



(c) Mesh 4, FEM (d) Mesh 4, NEFEM

Figure F.15: Inviscid subsonic flow around a NACA0012 airfoil: detail of the entropy error for FEM and NEFEM

steady-state solution. Even if the mesh is drastically refined an entropy production is clearly observed near the curved wall. The $\mathcal{L}^2$ norm of the entropy error on the airfoil profile is 1.3×10^{-3} in the finest mesh. The exact boundary representation considered in NEFEM represents an important reduction of the entropy generation near curved

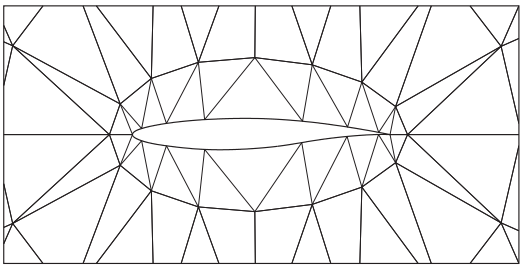
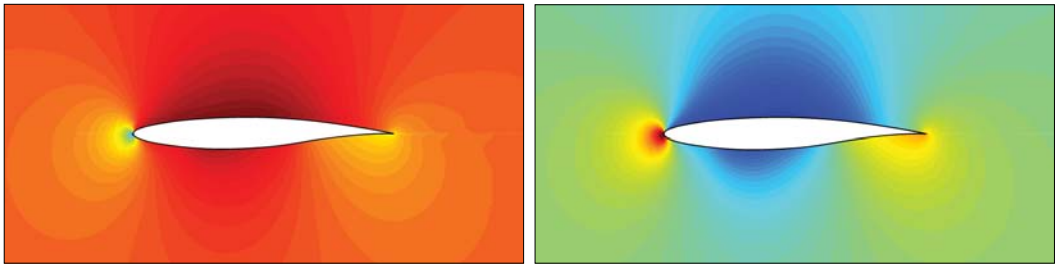


Figure F.16: Computational mesh for the computations of the inviscid subsonic flow around a RAE2822 airfoil



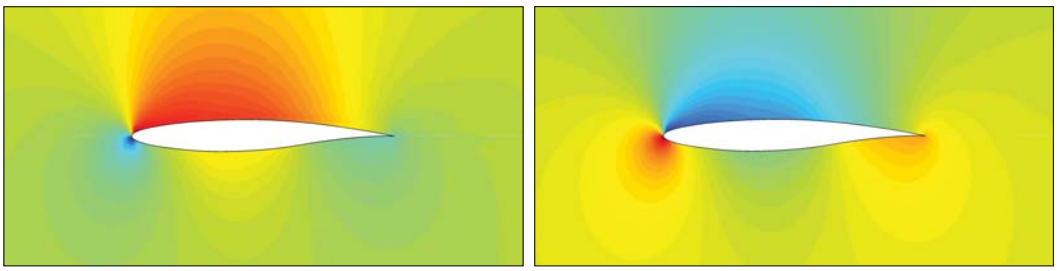
(a) Mach number (b) Pressure

Figure F.17: Inviscid subsonic flow around a RAE2822 airfoil at $M^\infty = 0.5$ and angle of attack 0 degrees: detail of the NEFEM solution with $p = 7$

walls. In the finest mesh, the $\mathcal{L}^2$ norm of the entropy error on the airfoil profile is 4×10^{-4}, more than two times more accurate than isoparametric FEs.

The subsonic flow around a RAE2822 airfoil is considered next. A subsonic flow at free-stream Mach number $M^\infty = 0.5$ and zero angle of attack is considered. A coarse mesh with only 14 curved elements to discretize the airfoil is considered, see Figure F.16, and high-order approximations are introduced to properly capture the solution. The mach number and pressure are depicted in Figure F.17, for a NEFEM solution with $p = 7$. The entropy error measured along the complete airfoil profile is 4.1×10^{-3}.

Finally, a subsonic flow at free-stream Mach number $M^\infty = 0.63$ and angle of attack of 2 degrees is considered. Figure F.18 shows two quantities of interest computed in the coarse mesh shown in Figure F.16, and by using a degree of approximation $p = 7$. The entropy error measured along the complete airfoil profile is 1.2×10^{-2},



(a) Mach number (b) Pressure

Figure F.18: Inviscid subsonic flow around a RAE2822 airfoil at $M^\infty = 0.63$ and angle of attack 2 degrees: detail of the NEFEM solution with $p = 7$

showing a good performance of NEFEM for more complex configurations.

Bibliography

Abarbanel, S. and D. Gottlieb (1998). On the construction of absorbing layers in CEM. *Appl. Numer. Math. 27*(4), 331–340.

Abarbanel, S., D. Gottlieb, and J. S. Hesthaven (2006). Non-linear PML equations for time dependent electromagnetics in three dimensions. *J. Sci. Comp. 28*(2-3), 125–137.

Ainsworth, M. (2004). Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods. *J. Comput. Phys. 198*(1), 106–130.

Amini, S. and S. M. Kirkup (1995). Solution of the Helmholtz equation in the exterior domain by elementary boundary integral methods. *J. Comp. Phys 118*(2), 208–221.

Anderson, J. D. (1982). *Modern Compressible Flow: with Historical Perspective.* New York: McGraw-Hill.

Arnold, D. N. (1982). An interior penalty finite element method with discontinuous elements. *SIAM J. Numer. Anal. 19*(4), 742–760.

Atkins, H. L. and C. W. Shu (1998). Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations. *AIAA Journal 36*(5), 775–782.

Babuška, I. and M. Suri (1987). The optimal convergence rate of the p-version of the finite element method. *SIAM J. Numer. Anal. 24*(4), 750–776.

Babuška, I., B. A. Szabó, and I. N. Katz (1981). The p-version of the finite element method. *SIAM J. Numer. Anal. 18*(3), 515–545.

Balanis, C. A. (1989). *Advanced Engineering Electromagnetics.* New York: John Wiley & Sons.

Banerjee, U. and M. Suri (1992). The effect of numerical quadrature in the p-version of the finite element method. *Math.Comp. 59*(199), 1–20.

Barth, T. and M. Ohlberger (2004). *Finite Volume Methods: Foundation and Analysis, in: E. Stein, R. de Borst, T.J.R. Hughes (Eds.), Fundamentals, Encyclopedia of Computational Mechanics, vol. 1.* New York: Wiley. Chapter 4.

Barth, T. J. (1998). *Simplified numerical methods for gas dynamics systems on triangulated domains*. Ph. D. thesis, Stanford University.

Bassi, F. and S. Rebay (1997). High-order accurate discontinuous finite element solution of the 2D Euler equations. *J. Comput. Phys. 138*(2), 251–285.

Bazilevs, Y., V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg (2008). Isogeometric analysis using T-splines. *Comput. Methods Appl. Mech. Engrg.*. in review.

Belytschko, T., A. Huerta, S. Fernández-Méndez, and T. Rabczuk (2004). *Meshfree methods*, Volume 1 (Fundamentals) of *Encyclopedia of Computational Mechanics*, Chapter 10. John Wiley & Sons.

Benson, T. (2008). Conformal mapping. `http://www.grc.nasa.gov/WWW/K-12/airplane/map.html`. [Online; accessed 21-May-2009].

Berenger, J. P. (1994). A perfectly matched layer for the absorption of electromagnetic waves. *J. Comp. Phys 114*(2), 185–200.

Bettess, P. (1992). *Infinite Elements*. Penshaaw Press.

Bonnet, F. and F. Poupaud (1997). Berenger absorbing boundary condition with time finite-volume scheme for triangular meshes. *Appl. Numer. Math. 25*(4), 333–354.

Brenner, S. C. and L. R. Scott (1994). *The Mathematical Theory of Finite Element Methods*. Springer.

Casoni, E., J. Peraire, and A. Huerta (2009). *ECCOMAS Multidisciplinary Jubilee Symposium*, Volume 14, Chapter One-Dimensional Shock-Capturing for High-Order Discontinuous Galerkin Methods, pp. 307–325. Springer.

Chen, M.-H., B. Cockburn, and F. Reitich (2005). High-order RKDG methods for computational electromagnetics. *J. Comp. Phys 22-23*(22-23), 205–226.

Chen, Q. and I. Babuška (1996). The optimal symmetrical points for polynomial interpolation of real functions in the tetrahedron. *Comput. Methods Appl. Mech. Engrg. 137*(1), 89–94.

Ciarlet, P. G. and P. A. Raviart (1972). Interpolation theory over curved elements, with applications to finite element methods. *Comput. Methods Appl. Mech. Engrg. 1*(1), 217–249.

Cioni, J. P., L. Fezoui, and H. Steve (1993). A parallel time-domain maxwell solver using upwind schemes and triangular meshes. *IMPACT Comput. Sci. Eng. 5*(3), 215–247.

Cirak, F., M. Ortiz, and P. Schröder (2000). Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *Internat. J. Numer. Methods Engng. 47*(12), 2039–2072.

Clough, R. W. (2004). Early history of the finite element method from the view point of a pioneer. *Internat. J. Numer. Methods Engng. 60*, 283–287.

Cockburn, B. (2004). *Discontinuous Galerkin methods for Computational Fluid Dynamics, in: E. Stein, R. de Borst, T.J.R. Hughes (Eds.), Fluids, Encyclopedia of Computational Mechanics, vol. 3.* New York: Wiley. Chapter 4.

Cockburn, B. and C. W. Shu (1989). TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation-laws II. General framework. *Math. Comp. 52*(186), 411–435.

Cockburn, B. and C. W. Shu (1998). The local Discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal. 35*(6), 2440–2463.

Costabel, M. and M. Dauge (1997). Singularities of electromagnetic fields in polyhedral domains. *Arch. Ration. Mech. Anal. 151*(3), 221–276.

Coyle, J. and P. D. Ledger (2005). Evidence of exponential convergence in the computation of Maxwell eigenvalues. *Comput. Methods Appl. Mech. Engrg. 194*(2-5), 587 – 604.

Demkowicz, L., J. T. Oden, W. Rachowicz, and O. Hardy (1989). Toward a universal *h-p* adaptive finite-element strategy. 1 Constrained approximation and data structure. *Comput. Methods Appl. Mech. Engrg. 77*(1-2), 79–112.

Dey, S., M. S. Shephard, and J. E. Flaherty (1997, December). Geometry representation issues associated with *p*-version finite element computations. *Comput. Methods Appl. Mech. Engrg. 150*(1-4), 39–55.

Dolejší, V. and M. Feistauer (2004). A semi-implicit discontinuous galerkin finite element method for the numerical solution of inviscid compressible flow. *J. Comput. Phys. 198*(2), 727–746.

Dominek, A. K. and H. T. Shamanski (1990). The almond test body. The Ohio State University ElectroScience Laboratory, Department of Electrical Engineering, Report 721929-9, NASA Langley Research Center.

Donea, J. and A. Huerta (2005). *Finite Element Methods for Flow Problems*. Wiley.

Fernández-Méndez, S. (2001). *Mesh-free methods and finite element: friends or foe?* Ph. D. thesis, Universitat Politènica de Catalunya.

Fernández-Méndez, S. and A. Huerta (2004). Imposing essential boundary conditions in mesh-free methods. *Comput. Methods Appl. Mech. Engrg. 192*(12-14), 1257–1275.

Ganesh, M. and I. G. Graham (2004). A high-order algorithm for obstacle scattering in three dimensions. *J. Comp. Phys 198*(1), 211–242.

Givoli, D. (1992). *Numerical Methods for Problems in Infinite Domains*. Amsterdam: Elsevier.

Gordon, W. J. and C. A. Hall (1973). Transfinite element methods: Blending-function interpolation over arbitrary curved element domains. *Numer. Math. 21*, 109–129.

Hachemi, M. E., O. Hassan, K. Morgan, D. Rowse, and N. Weatherill (2004). A low-order unstructured-mesh approach for computational electromagnetics in the time domain. *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 362*(1816), 445–469.

Harrington, R. F. (1961). *Time-Harmonic Electromagnetic Fields*. New York: McGraw-Hill.

Hesthaven, J. S. (2000). Stable spectral methods on tetrahedral elements. *SIAM J. Numer. Anal. 21*(6), 2352–2380.

Hesthaven, J. S. and T. Warburton (2002). Nodal high-order methods on unstructured grids I. time-domain solution of Maxwell's equations. *J. Comp. Phys 181*(1), 186–221.

Hillewaert, K., N. Chevaugeon, P. Geuzaine, and J.-F. Remacle (2006). Hierarchic multigrid iteration strategy for the discontinuous galerkin solution of the steady euler equations. *Internat. J. Numer. Methods Fluids 51*, 1157–1176.

Hirsch, C. (1988). *Numerical Computation of Internal and External Flows. Volume 1: Fundamentals of Numerical Discretization.* John Wiley & Sons.

Huerta, A., A. Rodríguez-Ferran, P. Díez, and J. Sarrate (1999). Adaptive finite element strategies based on error assessment. *Internat. J. Numer. Methods Engng. 46*, 1803–1818.

Hughes, T. J. R., J. A. Cottrell, and Y. Bazilevs (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Engrg. 194*(39–41), 4135–4195.

Huttunen, T., M. Malinen, and P. Monk (2007). Solving Maxwell's equations using the ultra weak variational formulation. *J. Comp. Phys 223*(2), 731–758.

Inoue, K., Y. Kikuchi, and T. Masuyama (2005). A NURBS finite element method for product shape design. *J. Engrg. Design 16*(2), 157–174.

Jiang, B.-N. (1998). *The Least-Squares Finite Element Method, Theory and Applications in Computational Fluid Dynamics and Electromagnetics*. Scientific Computation. Springer.

Jin, J.-M. and W. C. Chew (1996). Combining PML and ABC for the finite-element analysis of scattering problems. *Microwave Optical Technol. Lett. 12*(4), 192–197.

Johnson, C. (1987). *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge: Cambridge University Press.

Kabakian, A. V., V. Shankar, and W. F. Hall (2004). Unstructured grid-based discontinuous Galerkin method for broadband electromagnetic simulations. *J. Sci. Comp. 20*(3), 405–431.

Kagan, P., A. Fischer, and P. Z. Bar-Yoseph (1998). New B-spline finite element approach for geometrical design and mechanical analysis. *Internat. J. Numer. Methods Engng. 41*, 435–458.

Kangro, U. and R. Nicolaides (1997). Spurious fields in time-domain computations of scattering problems. *IEEE Trans. Antennas Propagat. 45*(2), 228–234.

Krivodonova, L. and M. Berger (2006, January). High-order accurate implementation of solid wall boundary conditions in curved geometries. *J. Comput. Phys. 211*(2), 492–512.

Ladson, C. L., C. W. Brooks, A. S. Hill, and D. W. Sproles (1996). Computer program to obtain ordinates for naca airfoils. Technical Report NASA TM-4741, NASA Langley Research Center.

Laney, C. B. (1998). *Computational gasdynamics*. Cambridge: Cambridge University Press.

Ledger, P. D. (2001). *An hp adaptive finite element proceedure for electromagnetic scattering problems*. Ph. D. thesis, Univ. Wales Swansea.

Ledger, P. D. and K. Morgan (2005). The application of the *hp*–finite element method to electromagnetic problems. *Arch. Comput. Methods Engrg. 12*(3), 235–302.

Ledger, P. D., K. Morgan, and O. Hassan (2007). Electromagnetic scattering simulation using an **H**(curl) conforming *hp* finite element method in three dimensions. *Internat. J. Numer. Methods Fluids 53*(8), 1267–1296.

Ledger, P. D., K. Morgan, O. Hassan, and N. P. Weatherill (2003). Plane wave h(curl; omega) conforming finite elements for Maxwell's equations. *Computat. Mech. 31*(3-4), 272–283.

Lenoir, M. (1986). Optimal isoparametric finite elements and error estimates for domains involving curved boundaries. *SIAM J. Numer. Anal. 23*(3), 562–580.

LeVeque, R. J. (1992). *Numerical methods for conservation laws* (Second ed.). Lectures in Mathematics ETH Zürich. Basel: Birkhäuser Verlag.

LeVeque, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press.

Luo, X.-J., M. S. Shephard, and J.-F. Remacle (2001). Influence of geometric approximation on the accuracy of higher order methods. Technical Report 1, SCOREC.

Luo, X.-J., M. S. Shephard, and J.-F. Remacle (2002). The influence of geometric approximation on the accuracy of higher order methods. In *8th International Conference on Numerical Grid Generation in Computational Field Simulations*.

Luo, X.-J., M. S. Shephard, J.-F. Remacle, R. M. O'Bara, M. W. Beall, B. Szabó, and R. Actis (2002). *p*-version mesh generation issues. In *11th International Meshing Roundtable*, pp. 343–354. Sandia National Laboratories.

Mäkipelto, J. (2004). Geometry based rational enrichment functions for triangular plane elasticity element. In *Proceedings of the 21st International Congress of Theoretical and Applied Mechanics*, Poland.

Montlaur, A. (2009). *High order Discontinuous Galerkin methods for incompressible flows*. Ph. D. thesis, Universitat Politècnica de Catalunya.

Montlaur, A., S. Fernández-Méndez, and A. Huerta (2008). Discontinuous galerkin methods for the stokes equations using divergence-free approximations. *Internat. J. Numer. Methods Fluids 57*(9), 1071–1092.

Morgan, K., O. Hassan, N. E. Pegg, and N. P. Weatherill (2000). The simulation of electromagnetic scattering in piecewise homogeneous media using unstructured grids. *Computat. Mech. 25*, 438–447.

Muñoz, J. J. (2008). Modelling unilateral frictionless contact using the null-space method and cubic B-Spline interpolation. *Comput. Methods Appl. Mech. Engrg. 197*(9-12), 979–993.

NPARC Alliance (2008). Computational fluid dynamics CFD verification and validation web site. `http://www.grc.nasa.gov/WWW/wind/valid/validation.html`. [Online; accessed 21-May-2009].

Pascal, J. F. and P.-L. George (1999). *Maillages : applications aux éléments finis*. Paris: Hermès Science Publications.

Peraire, J. and P.-O. Persson (2008). The compact discontinuous Galerkin (CDG) method for elliptic problems. *SIAM J. Sci. Comput. 30*(4), 1806–1824.

Persson, P.-O. and J. Peraire (2006a). An efficient low memory implicit DG algorithm for time dependent problems. In *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA.

Persson, P.-O. and J. Peraire (2006b). Sub-cell shock capturing for discontinuous Galerkin Methods. In *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA.

Persson, P.-O. and J. Peraire (2009). Curved mesh generation and mesh refinement using lagrangian solid mechanics. In *Proceedings of the 47th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA.

Piegl, L. and W. Tiller (1995). *The NURBS Book*. London: Springer-Verlag.

Polycarpou, A. C., M. R. Lyons, and C. A. Balanis (1996). A two-dimensional finite element formulation of the perfectly matched layer. *IEEE Microwave and Guided Wave Letters 6*(9), 338–340.

Remacle, J.-F., N. Chevaugeon, E. Marchandise, and C. Geuzaine (2005). Efficient visualization of high order finite elements. *Internat. J. Numer. Methods Engrg. 69*(4), 750–771.

Roca, X. (2009). *Paving the path towards automatic hexahedral mesh generation*. Ph. D. thesis, Universitat Politènica de Catalunya.

Sármány, D., M. A. Botchev, and J. J. W. van der Vegt (2007). Dispersion and dissipation error in high-order runge-kutta discontinuous galerkin discretisations of the maxwell equations. *J. Sci. Comp. 33*(1), 47–74.

Schramm, U. and W. Pilkey (1993). The coupling of geometric descriptions and finite elements using NURBS: a study of shape optimization. *Finite Elem. Anal. Des. 15*(1), 11–34.

Schuh, M. J., A. C. Woo, and M. P. Simon (1994). The monostatic/bistatic approximation. *IEEE Trans. Antennas Propagat. Mag. 35*(4), 76–78.

Scott, L. R. (1973). *Finite element techniques for curved boundaries*. Ph. D. thesis, Massachusetts Institute of Technology, Dept. of Mathematics.

Selig, M. (2008). UIUC airfoil database. `http://www.ae.uiuc.edu/m-selig/ads/coord_database.html`. [Online; accessed 21-May-2009].

Sevilla, R., S. Fernández-Méndez, and A. Huerta (2008a). NURBS-enhanced finite element method (NEFEM). *Internat. J. Numer. Methods Engrg. 76*(1), 56–83.

Sevilla, R., S. Fernández-Méndez, and A. Huerta (2008b). NURBS-enhanced finite element method (NEFEM) for Euler equations. *Internat. J. Numer. Methods Fluids 57*(9), 1051–1069.

Sevilla, R. and S. Fernández-Méndez (2009). Numerical integration for the 2D NURBS-enhanced finite element method (NEFEM). *Finite Elem. Anal. Des.*. in review.

Szabó, B. and I. Babuška (1991). *Finite Element Analysis*. New York: John Wiley & Sons.

Szabó, B., A. Düster, and E. Rank (2004). *The p-version of the Finite Element Method*, Volume 1 (Fundamentals) of *Encyclopedia of Computational Mechanics*, Chapter 5. New York: Wiley.

Szegö, G. (1975). *Orthogonal Polynomials (fourth edition)*. Providence: Americam Mathematical Society.

Taflove, A. (1995). *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, Inc.

Taylor, M. A., B. A. Wingate, and R. E. Vincent (2000). An algorithm for computing fekete points in the triangle. *SIAM J. Numer. Anal. 38*(5), 1707–1720.

Toro, E. F. (1997). *Riemann solvers and numerical methods for fluid dynamics*. Berlin: Springer.

Turkel, E. and A. Yefet (1998). Absorbing PML boundary layers for wave-like equations. *Appl. Numer. Math. 27*(4), 533–557.

Van der Ven, H. and J. J. W. Van der Vegt (2002). Space-time Discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows II. Efficient flux quadrature. *Comput. Methods Appl. Mech. Engrg. 191*(41-42), 4747–4780.

Vinh, H., C. P. van Dam, and H. A. Dwyer (1997). Finite difference maxwell solver to study geometric shape effects on radar signature. *J. Aircraft 34*(1), 56–63.

Wachspress, E. L. (1973). A rational basis for function approximation. II Curved sides. *J. Inst. Math. Appl. 11*, 83–104.

Wandzura, S. and H. Xiao (2003). Symmetric quadrature rules on a triangle. *Comput. Math. Appl. 45*(12), 1829–1840.

Wang, Z. J. and Y. Liu (2006). Extension of the spectral volume method to high-order boundary representation. *J. Comput. Phys. 211*, 154–178.

Warburton, T. (2006). An explicit construction of interpolation nodes on the simplex. *J. Eng. Math. 56*(3), 247–262.

Woo, A. C., H. T. G. Wang, and M. J. Schub (1993). Benchmark radar targets for the validation of computational electromagnetics programs. *IEEE Trans. Antennas Propagat. Mag. 35*(1), 84–89.

Xue, D. and L. Demkowicz (2005). Control of geometry induced error in *hp* finite element (FE) simulations. I. Evaluation of FE error for curvilinear geometries. *Internat. J. Numer. Anal. Model. 2*(3), 283–300.

Zienkiewicz, O. C. (1971). *The Finite Element Method in Engineering Science* (second ed.). London: McGraw-Hill.

Zienkiewicz, O. C. and R. L.Taylor (2000). *The Finite Element Method* (Fifth ed.), Volume 1. The basis. Butterwort-Heinemann.

Zienkiewicz, O. C., J. Z. Zbu, and N. G. Gong (1989). Efficient and practical $h-p$-adaptive analysis procedures for the finite element method. *Internat. J. Numer. Methods Engng. 28*(4), 879–891.

Zlamal, M. (1973a). Curved elements in the finite element method. I. *SIAM J. Numer. Anal. 10*(1), 229–240.

Zlamal, M. (1973b). The finite element method in domains with curved boundaries. *Internat. J. Numer. Methods Engng. 5*(3), 367–373.